Signal Detection in
Non-Gaussian Noise

Springer-Verlag

APPENDIX A

Appendix A

Saleem A. Kassam

Signal Detection in Non-Gaussian Noise

Consulting Editor: John B. Thomas

With 38 Illustrations

Springer-Verlag
New York Berlin Heidelberg
London Paris Tokyo

PREFACE

This book contains a unified treatment of a class of problems of signal detection theory. This is the detection of signals in additive noise which is not required to have Gaussian probability density functions in its statistical description. For the most part the material developed here can be classified as belonging to the general body of results of *parametric* theory. Thus the probability density functions of the observations are assumed to be known, at least to within a finite number of unknown parameters in a known functional form. Of course the focus is on noise which is not Gaussian; results for Gaussian noise in the problems treated here become special cases. The contents also form a bridge between the classical results of signal detection in Gaussian noise and those of *nonparametric* and *robust* signal detection, which are not considered in this book.

Three canonical problems of signal detection in additive noise are covered here. These allow between them formulation of a range of specific detection problems arising in applications such as radar and sonar, binary signaling, and pattern recognition and classification. The simplest to state and perhaps the most widely studied of all is the problem of detecting a *completely known deterministic signal* in noise. Also considered here is the detection of a *random non-deterministic signal* in noise. Both of these situations may arise for observation processes of the low-pass type and also for processes of the band-pass type. Spanning the gap between the known and the random signal detection problems is that of detection of a deterministic signal with random parameters in noise. The important special case of this treated here is the detection of *phase-incoherent narrowband signals* in narrowband noise.

There are some specific assumptions that we proceed under throughout this book. One of these is that ultimately all the data which our detectors operate on are *discrete sequences* of observation components, as opposed to being continuous-time waveforms. This is a reasonable assumption in modern implementations of signal detection schemes. To be able to treat non-Gaussian noise with any degree of success and obtain explicit, canonical, and useful results, a more stringent assumption is needed. This is the *independence* of the discrete-time additive noise components in the observation processes. There do exist many situations under which this assumption is at least a good approximation.

With the same objective of obtaining explicit canonical results of practical appeal, this book concentrates on *locally optimum* and *asymptotically optimum* detection schemes. These criteria are appropriate in detection of weak signals (the low

Saleem A. Kassam
Department of Electrical Engineering
University of Pennsylvania
Philadelphia, PA 19104
USA

signal-to-noise-ratio case), for which the use of optimum detectors is particularly meaningful and necessary to extract the most in detection performance.

Most of the development given here has not been given detailed exposition in any other book covering signal detection theory and applications, and many of the results have appeared relatively recently in technical journals. In presenting this material it is assumed only that the reader has had some exposure to the elements of statistical inference and of signal detection in Gaussian noise. Some of the basic statistics background needed to appreciate the rest of the development is reviewed in Chapter 1. This book should be suitable for use in a first graduate course on signal detection, to supplement the classical material on signal detection in Gaussian noise. Chapters 2-4 may be used to provide a fairly complete introduction to the known signal detection problem. Chapters 5 and 6 are on the detection of narrowband known and phase-incoherent signals, respectively, and Chapter 7 is on random signal detection. A more advanced course on signal detection may also be based on this book, with supplementary material on nonparametric and robust detection if desired. This book should also be useful as a reference to those active in research, as well as to those interested in the application of signal detection theory to problems arising in practice.

The completion of this book has been made possible through the understanding and help of many individuals. My family has been most patient and supportive. My graduate students have been very stimulating and helpful. Prashant Gandhi has been invaluable in getting many of the figures ready. For the excellent typing of the drafts and the final composition, I am grateful to Drucilla Spanner and to Diane Griffiths. Finally, I would like to acknowledge the research support I have received from the Air Force Office of Scientific Research and the Office of Naval Research, which eventually got me interested in writing this book.

Saleem A. Kassam
University of Pennsylvania
Philadelphia, PA 19104

TABLE OF CONTENTS

ELEMENTS OF STATISTICAL HYPOTHESIS TESTING

1.1 Introduction

The signal processing problem which is the object of our study in this book is that of detecting the presence of a signal in noisy observations. Signal detection is a function that has to be implemented in a variety of applications, the more obvious ones being in radar, sonar, and communications. By viewing signal detection problems as problems of binary hypothesis testing in statistical inference, we get a convenient mathematical framework within which we can treat in a unified way the analysis and synthesis of signal detectors for different specific situations. The theory and results in mathematical statistics pertaining to binary hypothesis-testing problems are therefore of central importance to us in this book. In this first chapter we review some of these basic statistics concepts. In addition, we will find in this chapter some further results of statistical hypothesis testing with which the reader may not be as familiar, but which will be of use to us in later chapters.

We begin in Section 1.2 with a brief account of the basic concepts and definitions of hypothesis-testing theory, which leads to a discussion of most powerful tests and the Neyman-Pearson lemma in Section 1.3. In Section 1.4 this important result is generalized to yield the structures of locally optimum tests, which we will make use of throughout the rest of this book. Section 1.5 reviews briefly the Bayesian approach to construction of tests for hypotheses. We shall not be using the Bayesian framework very much except in Chapters 2 and 5, where we shall develop locally optimum Bayes' detectors for detection of known signals in additive noise.

In the last section of this chapter we will introduce a measure which we will make use of quite extensively in comparing the performances of different detectors for various signal detection problems in the following chapters. While we will give a more general discussion of the *asymptotic relative efficiency* and the *efficacy* in Section 1.6, these measures will be introduced and discussed in detail for the specific problem of detection of a known signal in additive noise in Section 2.4 of Chapter 2. Readers may find it beneficial to postpone study of Section 1.6 until after Chapter 2 has been read; they may then better appreciate the applicability of the ideas and results of this section.

1.2 Basic Concepts of Hypothesis Testing

Let $X = (X_1, X_2, ..., X_n)$ be a random vector of observations with joint probability density function (pdf) $f_X(x|\theta)$, where θ is a parameter of the density function. Any specific realization $x = (x_1, x_2, ..., x_n)$ of X will be a point in R^n, where R is the set of all real numbers. In binary hypothesis-testing problems we have to decide between one of two hypotheses, which we will label as H and K, about the pdf $f_X(x|\theta)$, given an observation vector in R^n. Let Θ be the set of all possible values of θ; we usually identify H with one subset Θ_H with θ values and K with a disjoint subset Θ_K, so that $\Theta = \Theta_H \cup \Theta_K$. This may be expressed formally as

H: X has pdf $f_X(x|\theta)$ with $\theta \in \Theta_H$ (1-1)

K: X has pdf $f_X(x|\theta)$ with $\theta \in \Theta_K$ (1-2)

If Θ_H and Θ_K are made up of single elements, say θ_H and θ_K, respectively, we say that the hypotheses are *simple*; otherwise, the hypotheses are *composite*. If Θ can be viewed as a subset of R^p for a finite integer p, the pdf $f_X(x|\theta)$ is completely specified by the finite number p of real components of θ, and we say that our hypotheses are *parametric*.

A *test* for the hypothesis H against K may be specified as a partition of the sample space $S = R^n$ of observations into disjoint subsets S_H and S_K, so that x falling in S_H leads to acceptance of H, with K accepted otherwise. This may also be expressed by a *test function* $\delta(x)$ which is defined to have value $\delta(x) = 1$ for $x \in S_K$ and value $\delta(x) = 0$ for $x \in S_H$. The value of the test function is defined to be the probability with which the hypothesis K, the *alternative hypothesis*, is accepted. The hypothesis H is called the *null hypothesis*.

More generally, the test function can be allowed to take on probability values in the closed interval [0,1]. A test based on a test function taking on values inside [0,1] is called a *randomized test*.

The *power function* $p(\theta|\delta)$ of a test based on a test function δ is defined for $\theta \in \Theta_H \cup \Theta_K$ as

$$p(\theta|\delta) = E\{\delta(X)|\theta\}$$
$$= \int_{R^n} \delta(x) f_X(x|\theta)\, dx$$
(1-3)

Thus it is the probability with which the test will accept the alternative hypothesis K for any particular parameter value θ. When

θ is in Θ_H, the value of $p(\theta|\delta)$ gives the probability of an error, that of accepting K when H is correct. This is called a *type I error*, and depends on the particular value of θ in Θ_H. The *size* of a test is the quantity

$$\alpha = \sup_{\theta \in \Theta_H} p(\theta|\delta)$$
(1-4)

which may be considered as being the best upper bound on the type I error probability of the test.

In signal detection the null hypothesis is often a noise-only hypothesis, and the alternative hypothesis expresses the presence of a signal in the observations. For a detector D implementing a test function $\delta(x)$ the power function evaluated for any θ in Θ gives a *probability of detection* of the signal. Thus in later chapters we will use the notation $p_d(\theta|D)$ for the power function of a detector D, and in discussing the probability of detection at a particular value of the parameter θ in Θ_K (or for a simple alternative hypothesis K) we will use for it the notation p_d. The size of a detector is often called its *false-alarm probability*. This usage is encountered specifically when the noise-only null hypothesis is simple, and the notation for this probability is p_f.

1.3 Most Powerful Tests and the Neyman-Pearson Lemma

Given a problem of binary hypothesis testing such as defined by (1-1) and (1-2), the question arises as to how one may define and then construct an optimum test. Ideally, one would like to have a test for which the power function $p(\theta|\delta)$ has values close to zero for θ in Θ_H, and has values close to unity for θ in Θ_K. These are, however, conflicting requirements. We can instead impose the condition that the size α of any acceptable test be no larger than some reasonable level α_0, and subject to this condition look for a test for which $p(\theta|\delta)$, evaluated at a particular value θ_K of θ in Θ_K, has its largest possible value. Such a test is *most powerful* at level α_0 in testing H against the simple alternative $\theta = \theta_K$ in Θ_K; its test function $\delta^*(x)$ satisfies

$$\sup_{\theta \in \Theta_H} p(\theta|\delta^*) \leq \alpha_0$$
(1-5)

and

$$p(\theta_K|\delta^*) \geq p(\theta_K|\delta)$$
(1-6)

for all other test functions $\delta(x)$ of size less than or equal to α_0. In most cases of interest a most powerful level α_0 test satisfies (1-5)

with equality, so that its size is $\alpha = \alpha_0$.

For a simple null hypothesis H when $\theta = \theta_H$ is the only parameter value in Θ_H, the condition (1-5) becomes $p(\theta_H | \delta') \leq \alpha_0$ or $p_f \leq \alpha_0$, subject to which p_d at $\theta = \theta_K$ is maximized. For this problem of testing a simple H against a simple K, a fundamental result of Neyman and Pearson (called the Neyman-Pearson lemma) gives the structure of the most powerful test. We state the result here as a theorem:

Theorem 1: Let $\delta(x)$ be a test function of the form

$$\delta(x) = \begin{cases} 1 & , \quad \int x(x|\theta_K) > t\int x(x|\theta_H) \\ r(x) & , \quad \int x(x|\theta_K) = t\int x(x|\theta_H) \\ 0 & , \quad \int x(x|\theta_K) < t\int x(x|\theta_H) \end{cases} \qquad (1\text{-}7)$$

for some constant $t \geq 0$ and some function $r(x)$ taking on values in [0,1]. Then the resulting test is most powerful at level equal to its size for $H: \theta = \theta_H$ versus $K: \theta = \theta_K$.

In addition to the above sufficient condition for a most powerful test it can be shown that conversely, if a test is known to be most powerful at level equal to its size, then its test function must be of the form (1-7) except perhaps on a set of x values of probability measure zero. Additionally, we may always require $r(x)$ in (1-7) to be a constant r in [0,1]. Finally, we note that we are always guaranteed the existence of such a test for H versus K, of given size α [Lehmann, 1959, Ch. 3].

From the above result we see that generally the structure of a most powerful test may be described as one comparing a likelihood ratio to a constant threshold,

$$\frac{\int x(x|\theta_K)}{\int x(x|\theta_H)} > t \qquad (1\text{-}8)$$

in deciding if the alternative K is to be accepted. If the likelihood ratio on the left-hand side of (1-8) equals the threshold value t, the alternative K may be accepted with some probability r (the randomization probability). The constants t and r may be evaluated to obtain a desired size α using knowledge of the distribution function of the likelihood ratio under H.

When the alternative hypothesis K is composite we may look for a test which is *uniformly most powerful* (UMP) in testing H against K, that is, one which is most powerful for H against each $\theta = \theta_K$ in Θ_K. While UMP tests can be found in some cases,

notably in many situations involving Gaussian noise in signal detection, such tests do not exist for many other problems of interest. One option in such situations is to place further restrictions on the class of acceptable or admissible tests in defining a most powerful test; for example, a requirement of *unbiasedness* or of *invariance* may be imposed [Lehmann, 1959, Ch. 4-6]. As an alternative, other performance criteria based on the power function may be employed. We will consider one such criterion, leading to *locally optimum* or *locally most powerful* tests for composite alternatives, in the next section. One approach to obtaining reasonable tests for composite hypotheses is to use maximum-likelihood estimates $\hat\theta_H$ and $\hat\theta_K$ of the parameter θ, obtained under the constraints $\theta \in \Theta_H$ and $\theta \in \Theta_K$, respectively, in place of θ_H and θ_K in (1-8). The resulting test is called a *generalized likelihood ratio test* or simply a *likelihood ratio test* (see, for example, [Bickel and Doksum, 1977, Ch. 6]).

1.4 Local Optimality and the Generalized Neyman-Pearson Lemma

Let us now consider the approach to construction of tests for composite alternative hypotheses which we will use almost exclusively in the rest of our development on signal detection in non-Gaussian noise. In this approach attention is concentrated on alternatives $\theta = \theta_K$ in Θ_K which are close, in the sense of a metric or distance, to the null-hypothesis parameter value $\theta = \theta_H$. Specifically, let θ be a real-valued parameter with value $\theta = \theta_0$ defining the simple null hypothesis and let $\theta > \theta_0$ define the composite alternative hypothesis. Consider the class of all tests based on test functions $\delta(x)$ of a particular desired size α for $\theta = \theta_0$ against $\theta > \theta_0$, and assume that the power functions $p(\theta|\delta)$ of these tests are continuous and also continuously differentiable at $\theta = \theta_0$. Then if we are interested primarily in performance for alternatives which are close to the null hypothesis, we can use as a measure of performance the slope of the power function at $\theta = \theta_0$, that is,

$$p'(\theta_0|\delta) = p'(\theta|\delta)|_{\theta=\theta_0}$$
$$= \frac{d}{d\theta} p(\theta|\delta)\Big|_{\theta=\theta_0} \qquad (1\text{-}9)$$

From among our class of tests of size α, the test based on $\delta'(x)$ which uniquely maximizes $p'(\theta_0|\delta)$ has a power function satisfying

$$p(\theta|\delta') \geq p(\theta|\delta), \quad \theta_0 < \theta < \theta_{max} \qquad (1\text{-}10)$$

for some $\theta_{max} > \theta_0$. Such a test is called a *locally most powerful* or *locally optimum* (LO) test for $\theta = \theta_0$ against $\theta > \theta_0$. It is clearly of interest in situations such as the weak-signal case in signal detection, when the alternative-hypothesis parameter values of primary concern are those which define pdf's $f_X(x|\theta)$ close to the null-hypothesis noise-only pdf $f_X(x|\theta_H)$).

The following generalization of the Neyman-Pearson fundamental result of Theorem 1 can be used to obtain the structure of an LO test:

Theorem 2: Let $g(x)$ and $h_1(x), h_2(x), ..., h_m(x)$ be real-valued and integrable functions defined on R^n. Let an integrable function $\delta(x)$ on R^n have the characteristics

$$\delta(x) = \begin{cases} 1 & , \quad g(x) > \sum_{i=1}^{m} t_i h_i(x) \\ r(x) & , \quad g(x) = \sum_{i=1}^{m} t_i h_i(x) \\ 0 & , \quad g(x) < \sum_{i=1}^{m} t_i h_i(x) \end{cases}$$ (1-11)

for a set of constants $t_i \geq 0$, $i = 1,2,...,m$, and where $0 \leq r(x) \leq 1$. Define, for $i = 1,2,...,m$, the quantities

$$\alpha_i = \int_{R^n} \delta(x) h_i(x) \, dx$$ (1-12)

Then from within the class of all test functions satisfying the m constraints (1-12), the function $\delta(x)$ defined by (1-11) maximizes $\int_{R^n} \delta(x) g(x) \, dx$.

A more complete version of the above theorem, and its proof, may be found in [Lehmann, 1959, Ch. 3]; Ferguson [1967, Ch. 5] also discusses the use of this result.

To use the above result in finding an LO test for $\theta = \theta_0$ against $\theta > \theta_0$ defining Θ_H and Θ_K in (1-1) and (1-2), respectively, let us write (1-0) explicitly as

$$p'(\theta_0|\delta) = \frac{d}{d\theta} \int_{R^n} \delta(x) f_X(x|\theta) \, dx \Big|_{\theta=\theta_0}$$

$$= \int_{R^n} \delta(x) \frac{d}{d\theta} f_X(x|\theta) \Big|_{\theta=\theta_0} dx$$ (1-13)

assuming that our pdf's are such as to allow the interchange of the order in which limits and integration operations are performed. Taking $m=1$ and identifying $h_1(x)$ with $f_X(x|\theta_0)$ and $g(x)$ with $\frac{d}{d\theta} f_X(x|\theta)\Big|_{\theta=\theta_0}$ in Theorem 2, we are led to the locally optimum test which accepts the alternative $K: \theta > \theta_0$ when

$$\frac{\frac{d}{d\theta} f_X(x|\theta)\Big|_{\theta=\theta_0}}{f_X(x|\theta_0)} > t$$ (1-14)

where t is the test threshold value which results in a size-α test satisfying

$$E\{\delta(X)|H: \theta = \theta_0\} = \alpha$$ (1-15)

The test of (1-14) may also be expressed as one accepting the alternative when

$$\frac{d}{d\theta} \ln\{f_X(x|\theta)\}\Big|_{\theta=\theta_0} > t$$ (1-16)

Theorem 2 may also be used to obtain tests maximizing the *second derivative* $p''(\theta_0|\delta)$ at $\theta = \theta_0$. This would be appropriate to attempt if it so happens that $p'(\theta_0|\delta) = 0$ for all size-α tests for a given problem. The condition $p'(\theta_0|\delta) = 0$ will occur if $\frac{d}{d\theta} f_X(x|\theta)\Big|_{\theta=\theta_0}$ is zero, assuming the requisite regularity conditions mentioned above. In this case Theorem 2 can be applied to obtain the locally optimum test accepting the alternative hypothesis $K: \theta > \theta_0$ when

$$\frac{\frac{d^2}{d\theta^2} f_X(x|\theta)\Big|_{\theta=\theta_0}}{f_X(x|\theta_0)} > t$$ (1-17)

One type of problem for which Theorem 2 is useful in characterizing locally optimum tests is that of testing $\theta = \theta_0$ against the *two-sided* alternative hypothesis $\theta \neq \theta_0$. We have previously mentioned that one can impose the condition of *unbiasedness* on the allowable tests for a problem. Unbiasedness of a size-α test for the hypotheses H and K of (1-1) and (1-2) means that the test satisfies

$$P(\theta|\delta) \le \alpha, \text{ all } \theta \in \Theta_H \tag{1-18}$$

$$P(\theta|\delta) \ge \alpha, \text{ all } \theta \in \Theta_K \tag{1-19}$$

so that the detection probability for any $\theta_K \in \Theta_K$ is never less than the size α. For the two-sided alternative hypothesis $\theta \ne \theta_0$, suppose the pdf's $f_x(x|\theta)$ are sufficiently regular so that the power functions of all tests are twice continuously differentiable at $\theta = \theta_0$. Then it follows that for any *unbiased size-α test* we will have $p(\theta_0|\delta) = \alpha$ and $p'(\theta_0|\delta) = 0$. Thus, the test function of a locally optimum unbiased test can be characterized by using these two constraints and maximizing $p''(\theta_0|\delta)$ in Theorem 2. Another interpretation of the above approach for the two-sided alternative hypothesis is that the quantity $\omega = (\theta - \theta_0)^2$ may then be used as a measure of the distance of any alternative hypothesis from the null hypothesis $\theta = \theta_0$. We have

$$\frac{d}{d\omega} p(\theta|\delta)\bigg|_{\omega=0} = \frac{1}{2(\theta-\theta_0)} \frac{d}{d\theta} p(\theta|\delta)\bigg|_{\theta=\theta_0}$$

$$= \frac{1}{2} p''(\theta_0|\delta) \tag{1-20}$$

if $p'(\theta_0|\delta)$ is zero, for sufficiently regular pdf's $f_x(x|\theta)$. Thus if $p'(\theta_0|\delta)$ is zero for a class of size-α tests, then maximization of $p''(\theta_0|\delta)$ leads to a test which is locally optimum within that class.

1.5 Bayes Tests

In general statistical decision theory which can treat estimation and hypothesis-testing problems within a single framework, there are four fundamental entities. These are (a) the observation space, which in our case is R^n; (b) the set Θ of values of θ which parameterizes the possible distributions of the observations; (c) the set of all actions a which may be taken, the action space A; and (d) the loss function $l(\theta, a)$, a real-valued function measuring the loss suffered due to an action a when $\theta \in \Theta$ is the parameter value. In a binary hypothesis-testing problem the action space A will have only two possible actions, a_H and a_K, which, respectively, represent acceptance of the hypotheses H and K; and as a reasonable choice of loss function we can take

$$l(\theta, a) = c_{IJ}, \quad \theta \in \Theta_J, \text{ and } a = a_L \tag{1-21}$$

where $J, L = H$ or K, the c_{IJ} are non-negative, and $c_{IJ} = 0$. What is sought is a decision rule $d(x)$ taking on values in A, which specifies the action to be taken when an observation x has been made. More generally we can permit randomized decisions $\delta(x)$ which for each x specify a probability distribution over A.

The performance of any decision rule $d(x)$ can be characterized by the average loss that is incurred in using it; this is the *risk function*

$$R(\theta, d) = E\{l(\theta, d(x))|\theta\}$$

$$= \int_{R^n} l(\theta, d(x)) f_x(x|\theta) \, dx \tag{1-22}$$

The risk function for any given decision rule is nonetheless a function of θ, so that a comparison of the performances of different decision rules over a set of values of θ is not quite straightforward. A single real number serving as a figure of merit is assigned to a decision rule in Bayesian decision theory; to do this there is assumed to be available information leading to an *a priori* characterization of a probability distribution over Θ. We will denote the corresponding pdf as $\pi(\theta)$, and obtain the *Bayes risk* for a given prior density $\pi(\theta)$ and a decision rule $d(x)$ as

$$r(\pi, d) = E\{R(\theta, d)\}$$

$$= \int_{\Theta} R(\theta, d)\pi(\theta) \, d\theta \tag{1-23}$$

In the binary hypothesis-testing problem of deciding between Θ_H and Θ_K for θ in $f_x(x|\theta)$ [Equations (1-1) and (1-2)], the prior pdf $\pi(\theta)$ may be obtained as

$$\pi(\theta) = \pi_H \pi(\theta|H) + \pi_K \pi(\theta|K) \tag{1-24}$$

where π_H and π_K are the respective *a priori* probabilities that H and K are true $(\pi_H + \pi_K = 1)$, and $\pi(\theta|H)$ and $\pi(\theta|K)$ are the conditional *a priori* pdf's over θ, conditioned, respectively, on H and K being true. For the loss function of (1-21) this gives

$$r(\pi, d) = \int_{\Theta} \int_{R^n} l(\theta, d(x)) f_x(x|\theta)\pi(\theta) \, dx \, d\theta$$

$$= \int_{R^n} \int_{\Theta} l(\theta, d(x)) f_x(x|\theta) \left\{ \sum_{J=H,K} \pi_J \pi(\theta|J) \right\} d\theta \, dx \tag{1-25}$$

For observations x for which $d(x) = a_H$, the inner integral over Θ becomes

$$\pi_K c_{HK} \int_{\Theta_K} \int f_X(x|\theta)\pi(\theta|K)\, d\theta$$
$$= \pi_K c_{HK} \int f_X(x|K) \qquad (1\text{-}26)$$

where $f_X(x|K)$ is the conditional pdf of X given that the alternative hypothesis K is true. Similarly, for x such that $d(x) = a_K$ the integral over Θ in (1-25) becomes $\pi_K c_{KH} \int f_X(x|H)$. Thus the Bayes rule minimizing $r(\pi,d)$ accepts the alternative hypothesis K when

$$\frac{f_X(x|K)}{f_X(x|H)} > \frac{\pi_H c_{KH}}{\pi_K c_{HK}} \qquad (1\text{-}27)$$

When the likelihood ratio on the left-hand side above is strictly less than the threshold value on the right-hand side, the null hypothesis H is accepted by the Bayes rule. When equality holds $d(x)$ may specify any choice between a_H and a_K.

Note that for a test implementing Bayes' rule the threshold value is completely specified once the prior probabilities π_H and π_K are fixed; the false-alarm probability does not enter as a consideration in setting the threshold. The development above assumed that $\pi(\theta|K)$ and $\pi(\theta|H)$ are available when the hypotheses are composite. As an alternative approach when Θ_K is composite, one can consider *locally optimum Bayes'* tests for which only the first few terms in a power series expansion of the likelihood ratio $\int f_X(x|\theta_K)/\int f_X(x|\theta_H)$ are used on the left-hand side of (1-27). Such an approach has been considered by Middleton [1966, 1984] and we will describe its application in a detection problem in Section 2.5 of the next chapter.

1.6 A Characterization of the Relative Performance of Tests

In this final section before we proceed to consider signal detection problems we will develop a relative-performance measure which is conveniently applied and which is at the same time of considerable value in obtaining simple quantitative comparisons of different tests or detectors. This measure is called the *asymptotic relative efficiency* and we will use it, together with a detector performance measure related to it called the *efficacy*, quite extensively in this book.

Let D_A and D_B be two detectors based on test statistics $T_A(X)$ and $T_B(X)$, respectively, so that the test function $\delta_A(X)$ of

D_A is defined by

$$\delta_A(x) = \begin{cases} 1\,, & T_A(x) > t_A \\ r_A\,, & T_A(x) = t_A \\ 0\,, & T_A(x) < t_A \end{cases} \qquad (1\text{-}28)$$

with a similar definition for $\delta_B(x)$, in terms of respective thresholds t_A and t_B and randomization probabilities r_A and r_B. Suppose the thresholds t_A and t_B (and the randomization probabilities r_A and r_B, if necessary) are designed so that both detectors have the same size or false-alarm probability $p_f = \alpha$. To compare the relative detection performance or power of the two detectors one would have to obtain the power functions of the two detectors, that is, detection probabilities would have to be computed for all $\theta \in \Theta_K$. Furthermore, such a comparison would only be valid for one particular value of the size, α, so that power functions would have to be evaluated at all other values of the size which may be of interest. In addition, while it has not yet been made very explicit, such a comparison can be expected to depend on the number of observation components n used by the detectors when n is a design parameter; for example, the observation components X_i, $i=1,2,...,n$, may be identically distributed outcomes of repeated independent observations, forming a random sample of size n, governed by a univariate pdf $f_X(x|\theta)$.

In any event one would still be faced with the problem of expressing in a succinct and useful way the result of such a performance comparison. Of particular interest would be a real-valued measure which could be taken as an index of the overall relative performance of two tests or detectors. It would be even more appealing if such a measure could be computed directly from general formulas which would not require any explicit computation of families of power functions. As a step toward the definition of such a single index of relative performance, let us discuss a measure which is called the *relative efficiency* of two tests.

1.6.1 Relative Efficiency

Suppose that the number of observation components n in X is a variable quantity, so that any given detector has a choice of observation-vector size n that it can operate on for some given hypothesis-testing problem, namely testing $\theta \in \Theta_H$ against $\theta \in \Theta_K$ for θ in $\int f_X(x|\theta)$. For example, the linear detector test statistic is $\sum_{i=1}^{n} X_i$, and such a detector may be designed to operate on any sample size n. The situation which generally holds is that increasing n leads to improvement in the performance of a detector; for a fixed design value of the size α, for example, the detection

probability or power at any value of $\theta \in \Theta_K$ will increase with n. It is also generally true that there is a cost associated with the use of a larger number of observations, which in signal detection applications may mean a higher sampling rate, a longer observation time interval and therefore longer delay before a decision is made, or a heavier computational burden. Thus one measure of the relative performance of two tests or detectors in any given hypothesis-testing problem is obtained as a ratio of observation sizes required by the two procedures to attain a given level of performance.

Consider any specific alternative hypothesis defined by some θ_K in Θ_K, and let us consider detectors for $\theta \in \Theta_H$ against $\theta \in \Theta_K$ which are of size α and have detection probability $p_d = 1 - \beta$ for $\theta = \theta_K$, where the specified α and "miss" probability β are between 0 and 1. Let D_A and D_B be two detectors achieving this performance specification, based on respective observation vector lengths n_A and n_B. These observation vector lengths n_A and n_B are clearly functions of α, β and θ_K, and are more explicitly expressed as $n_A(\alpha,\beta,\theta_K)$ and $n_B(\alpha,\beta,\theta_K)$, respectively.

The *relative efficiency* $RE_{A,B}$ of the detectors D_A and D_B is defined as the ratio

$$RE_{A,B} = \frac{n_B(\alpha,\beta,\theta_K)}{n_A(\alpha,\beta,\theta_K)} \qquad (1\text{-}29)$$

This obviously depends on α, β and θ_K in general. In addition to this dependence of the value of $RE_{A,B}$ on the operating point (α,β,θ_K), the computation of this quantity requires knowledge of the probability distribution functions of $T_A(X)$ and $T_B(X)$ for $\theta = \theta_K$. In order to alleviate this requirement, which is often difficult to meet, one could look at the asymptotic case where both n_A and n_B become large, with the expectation that the limiting distributions of the test statistics become Gaussian. It turns out that under the proper asymptotic formulation of the problem the relative efficiency converges to a quantity which is independent of α and β and is much easier to evaluate in practice.

1.6.2 Asymptotic Relative Efficiency and Efficacy

For $\theta = \theta_K$ a fixed parameter in the alternative hypothesis observation pdf $f_X(x|\theta)$, consider the sequence of observation-vector lengths $n=1,2,\ldots$. Almost without exception, tests of hypotheses used in applications such as signal detection have the property that, for fixed α, the power of the tests increase to a limiting value of unity (or $\beta \to 0$) for $n \to \infty$. Thus in seeking an asymptotic definition of relative efficiency this effect needs to be addressed. Let us first formalize this type of behavior by giving a

definition of the property of *consistency of tests*.

The H and K of (1-1) and (1-2) were stated for the pdf of a vector of length n of observations. Consider now a *sequence* $\{\theta_n, n = 1,2,\ldots\}$ of θ values in Θ_K, and the sequence of corresponding hypothesis-testing problems that we get for $n = 1,2,\ldots$:

$$H_n: \text{ X has pdf } f_X(x|\theta) \text{ with } \theta \in \Theta_H \qquad (1\text{-}30)$$

$$K_n: \text{ X has pdf } f_X(x|\theta_n) \qquad (1\text{-}31)$$

Let $\{T_n(X), n = 1,2,\ldots\}$ be the *sequence* of test statistics, with $X = (X_1, X_2, \ldots, X_n)$, of some detector type. This generally means that $T_n(X)$ for different n has a fixed functional form; for example, $T_n(X) = \sum_{i=1}^n X_i$. Now the sequence of individual detectors D_n based on the $T_n(X)$ (and corresponding threshold and randomization probabilities t_n and r_n, respectively) is *asymptotically of size* α in testing $\theta \in \Theta_K$ if

$$\lim_{n \to \infty} \; \sup_{\theta \in \Theta_H} E(\delta_n(X)|\theta) = \alpha \qquad (1\text{-}32)$$

Here $\delta_n(X)$ is the test function obtained using the statistic $T_n(X)$. Such an asymptotically size-α sequence $\{D_n, n=1,2,\ldots\}$ of detectors is said to be *consistent* for the sequence of alternatives $\{\theta_n, n=1,2,\ldots\}$ if

$$\lim_{n \to \infty} p(\theta_n \mid \delta_n) = \lim_{n \to \infty} p_d(\theta_n \mid D_n)$$
$$= 1 \qquad (1\text{-}33)$$

We have remarked above that if θ_n is fixed to be some θ_K in Θ_K, sequences of most of the types of tests of interest to us will be consistent. In order to consider asymptotic situations where observation lengths n become large in relative efficiency considerations, one approach therefore is to define sequences of alternatives parameterized by the θ_n, for which the limiting power is not unity but is a specified value $1-\beta$ (with $0 < \beta < 1$). This clearly implies that the θ_n should approach a null-hypothesis parameter value as $n \to \infty$. We have stated earlier that the local case in hypothesis testing, where null- and alternative-hypothesis distributions of the observations approach one another, is of considerable interest to us since in many signal detection problems the weak-signal case is of particular concern. Thus we should expect that such an asymptotic formulation of the hypothesis-testing problem will be quite useful, corresponding to a situation of a weak signal and a long observation record in signal detection.

Let us now focus on hypothesis-testing problems where the parameter θ is *real-valued*, and further constrain the null hypothesis to be *simple*. In particular, without loss of generality, let $\theta = 0$ be taken to describe the simple null hypothesis. Furthermore, let us consider the one-sided alternative hypothesis described by $\theta > 0$. We are now ready to show that an asymptotic version of the relative efficiency may be defined and computed quite readily for two tests, under some regularity conditions which are frequently satisfied.

Let $\{\theta_l, l=1,2,...\}$ be a sequence of alternative-hypothesis parameters converging to $\theta = 0$, and let $\{n_{A,l}\}$ and $\{n_{B,l}\}$ be two corresponding sequences of observation-vector lengths for two detector sequences $\{D_A^l\}$ and $\{D_B^l\}$, respectively, so that D_A^l uses $n_{A,l}$ observations X_i in testing $\theta = 0$ against $\theta = \theta_l$, and similarly, D_B^l uses an observation-vector length of $n_{B,l}$.

Definition: Let $\{D_A^l\}$ and $\{D_B^l\}$ be of asymptotic size α for $\theta = 0$, and let their limiting power values for $l \to \infty$ exist and be equal for the alternatives $\{\theta_l\}$,

$$\lim_{l\to\infty} P_l(\theta_l \mid D_A^l) = \lim_{l\to\infty} P_l(\theta_l \mid D_B^l)$$
$$= 1 - \beta \qquad (1\text{-}34)$$

with $0 < \beta < 1$. Then the *asymptotic relative efficiency* (ARE) of $\{D_A^l\}$ with respect to $\{D_B^l\}$ is defined by

$$ARE_{A,B} = \lim_{l\to\infty} \frac{n_{B,l}}{n_{A,l}} \qquad (1\text{-}35)$$

provided that the limiting value in (1-35) is independent of α and β and of the particular sequences $\{\theta_l\}$, $\{n_{A,l}\}$, and $\{n_{B,l}\}$ satisfying (1-34).

The above definition of the ARE is clearly an asymptotic version of the relative efficiency. The main feature of this asymptotic definition is that we have considered a sequence of alternative-hypothesis parameter values which approach the null-hypothesis value $\theta = 0$, and corresponding observation-vector lengths which grow in such a way that the *asymptotic* powers become some value $1 - \beta$ between zero and unity. Note that we also require the ARE to be independent of the asymptotic size α of the detectors and of β. As we have remarked before, the detectors D_A^l (or D_B^l) are for different l members of a particular type of procedure having a common functional dependence of its test statistic on X, and we often speak of such a whole sequence of detectors as a detector D_A (or D_B).

Regularity Conditions

Consider a sequence of detectors $\{D_n\}$ based on a corresponding sequence of test statistics $\{T_n(X)\}$ with threshold sequence $\{t_n\}$. Let the detector sequence be of asymptotic size α, and let $\mu_n(\theta)$ and $\sigma_n^2(\theta)$ be the mean and variance, respectively, of $T_n(X)$ for $\theta \geq 0$. The following regularity conditions make computation of the ARE quite simple in many cases:

(i) $\left. \dfrac{d}{d\theta} \mu_n(\theta) \right|_{\theta=0} = \mu_n'(0)$ exists and is positive;

(ii) $\displaystyle \lim_{n\to\infty} \frac{\mu_n'(0)}{\sqrt{n}\, \sigma_n(0)} = c > 0$; $\qquad (1\text{-}36)$

For $\theta_n = \gamma/\sqrt{n}$, with $\gamma \geq 0$,

(iii) $\displaystyle \lim_{n\to\infty} \frac{\mu_n'(\theta_n)}{\mu_n'(0)} = 1$ $\qquad (1\text{-}37)$

and

$$\lim_{n\to\infty} \frac{\sigma_n(\theta_n)}{\sigma_n(0)} = 1; \qquad (1\text{-}38)$$

(iv) $[T_n(X) - \mu_n(\theta_n)]/\sigma_n(\theta_n)$ has asymptotically the standard normal distribution.

If these regularity conditions are satisfied by two given detectors, we can show that their ARE can be computed quite easily. Notice that for conditions (iii) and (iv) above a particular rate of convergence of θ_n to zero is used, so that in making ARE evaluations using these conditions, we are assuming this type of convergence of θ_n to zero. We should also remark that in many cases which will be of interest to us condition (iv) may be shown to hold; this is usually true in particular when $X = (X_1, X_2, ..., X_n)$ is a vector of independent components and the test statistic $T_n(X)$ is the log of the likelihood ratio. The regularity conditions (i) - (iii) also hold for many test statistics of interest. The D_n being of asymptotic size α, it follows easily from condition (iv) that $[t_n - \mu_n(0)]/\sigma_n(0)$ converges to z_α where

$$\alpha = 1 - \Phi(z_\alpha) \qquad (1\text{-}39)$$

Φ being the standard normal distribution function. We also get from the regularity conditions

$$\lim_{n \to \infty} p_d(\theta_n \mid D_n) = \lim_{n \to \infty} p(T_n(X) > t_\alpha \mid \theta_n)$$

$$= 1 - \Phi(t_\alpha - \gamma c) \qquad (1\text{-}40)$$

where c was defined in (1-36) and $\theta_n = \gamma/\sqrt{n}$. The quantity c^2 is called the efficacy[1] ξ of the detector sequence $\{D_n\}$, which is therefore

$$\xi = \lim_{n \to \infty} \frac{\left[\dfrac{d}{d\theta} E\{T_n(X) \mid \theta\}\Big|_{\theta=0}\right]^2}{n \, V(T_n(X) \mid \theta)\Big|_{\theta=0}} \qquad (1\text{-}41)$$

From these results we get the following theorem:

Theorem 3: Let D_A and D_B be asymptotic size-α detector sequences whose test statistics satisfy the regularity conditions (i) - (iv). Then the $ARE_{A,B}$ of D_A relative to D_B is

$$ARE_{A,B} = \frac{\xi_A}{\xi_B} \qquad (1\text{-}42)$$

where ξ_A and ξ_B are the efficacies of D_A and D_B, respectively.

To prove this result, note that if $\gamma = \gamma_A > 0$ and $\gamma = \gamma_B > 0$ define the sequences of alternatives $\theta_n = \gamma/\sqrt{n}$ for D_A and D_B, respectively, then the detector sequences $\{D_{A,n}\}$ and $\{D_{B,n}\}$ have the same asymptotic power if

$$\gamma_A \xi_A^{1/2} = \gamma_B \xi_B^{1/2} \qquad (1\text{-}43)$$

On the other hand, consider two subsequences $\{D_A^l\}$ and $\{D_B^l\}$ from $\{D_{A,n}\}$ and $\{D_{B,n}\}$, respectively, such that asymptotically, for $l \to \infty$,

$$\frac{\gamma_A}{\sqrt{n_{A,l}}} = \frac{\gamma_B}{\sqrt{n_{B,l}}} \qquad (1\text{-}44)$$

where γ_A and γ_B satisfy (1-43). Then for $\theta_l = \gamma_A/\sqrt{n_{A,l}}$ both $\{D_A^l\}$ and $\{D_B^l\}$ have the same limiting power as $l \to \infty$, given by (1-40) with $\gamma_C = \gamma_A \xi_A^{1/2} = \gamma_B \xi_B^{1/2}$. From (1-43) and (1-44), and the definition (1-35), the result of Theorem 3 follows easily.

---

[1] The quantity c itself is usually called the efficacy in statistics.

Thus we find that, subject to the regularity conditions (i) - (iv), the ARE of two detectors can be computed quite simply as a ratio of their efficacies; the efficacy of a detector is obtained from (1-41), which can be seen to require derivation of the means and variances only of the test statistics. We will be basing most of our performance comparisons in this book on the ARE, so that in the following chapters we will apply Theorem 3 and (1-41) many times. In the next chapter we will enter into a further discussion of the ARE for the special case of detection of known signals in additive noise, which will serve to illustrate more explicitly the ideas we have developed here.

Extended Regularity Conditions

As a generalization of the above results, we can consider the following extended versions of the conditions (i) - (iv):

(i)' $\dfrac{d^i}{d\theta^i} \mu_n(\theta)\Big|_{\theta=0} = \mu_n^{(i)}(0)$

$$= 0, \quad i = 1,2,...,m-1,$$

and

$$\mu_n^{(m)}(0) > 0$$

for some integer m;

(ii)' $\displaystyle \lim_{n \to \infty} \frac{\mu_n^{(m)}(0)}{n^{m\delta} \sigma_n(0)} = c > 0 \qquad (1\text{-}45)$

for some $\delta > 0$;

For $\theta_n = \gamma/n^\delta$, with $\gamma \geq 0$,

(iii)' $\displaystyle \lim_{n \to \infty} \frac{\mu_n^{(m)}(\theta_n)}{\mu_n^{(m)}(0)} = 1 \qquad (1\text{-}46)$

and

$$\lim_{n \to \infty} \frac{\sigma_n(\theta_n)}{\sigma_n(0)} = 1; \qquad (1\text{-}47)$$

(iv)' $[T_n(X) - \mu_n(\theta_n)]/\sigma_n(\theta_n)$ has asymptotically the standard normal distribution.

If we now define the efficacy by

$$\xi = \lim_{n \to \infty} \frac{1}{n} \left[\frac{\mu_n^{(m)}(0)}{\sigma_n(0)} \right]^{1/m\delta}$$ (1-48)

the ARE of two detectors satisfying the regularity conditions (i)' - (iv)' for a common δ and the same value of m can again be obtained as a ratio of their efficacies. Note that $m=1$ and $\delta=1/2$ is the case we considered first; it is the most commonly occurring case.

While we have focused on the case of alternatives $\theta > 0$, it should be quite clear that for alternatives $\theta < 0$ exactly the same results hold, which can be established by reparameterizing with $\theta_c = -\theta$.

The extended results above also allow consideration of certain tests for the two-sided alternatives $\theta \neq 0$. For detailed discussions of the asymptotic comparisons of tests the reader may consult the book by Kendall and Stuart [1967]. The original investigation of this type of asymptotic performance evaluation was made by Pitman [1948] and generalized by Noether [1955].

The Multivariate Case

In the discussion above of the efficacy of a test and of the ARE of two tests we required that the distributions of the test statistics be asymptotically normal. A careful examination of the above development reveals, however, that it is not the asymptotic normality of the test statistics which is the key requirement. What is needed to obtain the ARE of two tests is that their test statistics have the *same* functional form for their asymptotic distributions, so that the asymptotic powers of the two tests based on them can be made the same by proper choice of the sample sizes. Later in this book we will consider test statistics which may be expressed as quadratic forms having asymptotically the χ^2 distribution. Let us therefore extend our definition of the efficacy of a test to allow us to treat such cases.

Consider a sequence $\{T_n(X)\}$ of *multivariate* statistics with

$$T_n(X) = [T_{1n}(X), T_{2n}(X),..., T_{Ln}(X)]$$ (1-49)

Let the means and variances of the components for $\theta \geq 0$ be

$$E\{T_{jn}(X) | \theta\} = \mu_{jn}(\theta), \quad j = 1,2,...,L$$ (1-50)
$$V\{T_{jn}(X) | \theta\} = \sigma_{jn}^2(\theta), \quad j = 1,2,...,L$$ (1-51)

and let the covariances for $\theta \geq 0$ be

$$COV\{T_{jn}(X) T_{kn}(X) | \theta\} = \rho_{kjn}(\theta) \sigma_{kn}(\theta) \sigma_{jn}(\theta)$$ (1-52)

so that $\rho_{kjn}(\theta)$ is the normalized covariance or coefficient of correlation. We now impose the multivariate versions of the extended regularity conditions given above for the scalar case:

(i)'' $$\frac{d^i}{d\theta^i} \mu_{jn}(\theta) \Big|_{\theta=0} = \mu_{jn}^{(i)}(0)$$
$$= 0, \quad i = 1,2,...,m-1$$

and
$$\mu_{jn}^{(m)}(0) > 0$$

for some integer m, for $j = 1,2,...,L$;

(ii)'' $$\lim_{n \to \infty} \frac{\mu_{jn}^{(m)}(0)}{n^m \sigma_{jn}(0)} = c_j > 0$$ (1-53)

for some $\delta > 0$, for $j = 1,2,...,L$;

For $\theta_n = \gamma/n^\delta$, with $\gamma \geq 0$,

(iii)'' $$\lim_{n \to \infty} \frac{\mu_{jn}^{(m)}(\theta_n)}{\mu_{jn}^{(m)}(0)} = 1$$ (1-54)

$$\lim_{n \to \infty} \frac{\sigma_{jn}(\theta_n)}{\sigma_{jn}(0)} = 1$$ (1-55)

and
$$\lim_{n \to \infty} \frac{\rho_{kjn}(\theta_n)}{\rho_{kjn}(0)} = 1$$ (1-56)

for $k,j = 1,2,...,L$;

(iv)'' the $[T_{jn}(X) - \mu_{jn}(\theta_n)]/\sigma_{jn}(\theta_n)$, for $j = 1,2,...,L$, have asymptotically a multivariate normal distribution.

Let us define the normalized versions $Q_{jn}(X,\theta)$ of the components $T_{jn}(X)$ of the multivariate statistics as

$$Q_{jn}(X,\theta) = \frac{T_{jn}(X) - \mu_{jn}(\theta)}{\sigma_{jn}(\theta)}, \quad j = 1,2,...,L \quad (1\text{-}57)$$

and let $Q_n(X,\theta)$ be the row vector of components $Q_{jn}(X,\theta)$. Now a real-valued statistic of particular interest is that defined by the quadratic form

$$U_n(X) = Q_n(X,0) R_n^{-1}(0) Q_n(X,0)^T \quad (1\text{-}58)$$

where the matrix $R_n(\theta)$ is the normalized $L \times L$ covariance matrix of elements $\rho_{kjn}(\theta)$. Under the null hypothesis $\theta = 0$ this statistic has asymptotically a χ^2 distribution with L degrees of freedom. To obtain its limiting distribution under the alternatives $\theta_n = \gamma/n^\delta$, note that $Q_{jn}(X,0)$ can be expressed as

$$Q_{jn}(X,0) = Q_{jn}(X,\theta_n) \frac{\sigma_{jn}(\theta_n)}{\sigma_{jn}(0)} + \frac{\mu_{jn}(\theta_n) - \mu_{jn}(0)}{\sigma_{jn}(0)} \quad (1\text{-}59)$$

so that it is asymptotically equivalent to $Q_{jn}(X,\theta_n) + (\gamma^m/m!)c_j$, where c_j was defined by (1-53) in (ii)''. Using condition (iv)'' we conclude finally that $U_n(X)$ has asymptotically a non-central χ^2 distribution with L degrees of freedom under the sequence of alternative hypotheses defined by $\theta_n = \gamma/n^\delta$. The non-centrality parameter in this distribution is

$$\Delta = \frac{\gamma^{2m}}{(m!)^2} c R^{-1} c^T \quad (1\text{-}60)$$

where $c = (c_1, c_2, ..., c_L)$ and $R = \lim_{n \to \infty} R_n(0)$.

Suppose we have two tests based on L-variate quantities $T_{A,n}(X)$ and $T_{B,n}(X)$ satisfying the multivariate extended regularity conditions (i)'' - (iv)''. Let their respective vectors c of individual efficacy components be denoted as c_A and c_B, and let their respective limiting normalized covariance matrices be R_A and R_B. Assume further that the conditions (i)'' - (iv)'' are satisfied for the same values of m and δ by both sequences of statistics, and consider the quadratic form test statistics defined by (1-58) for each

of the two tests. It follows exactly as in the scalar cases treated above that for the same sequence of alternatives $\theta_n = \gamma/n^\delta$, the $ARE_{A,B}$ of the two tests is

$$ARE_{A,B} = \left[\frac{c_A R_A^{-1} c_A^T}{c_B R_B^{-1} c_B^T} \right]^{1/(2m\delta)} \quad (1\text{-}61)$$

We may therefore define the generalized efficacy of a test based on a quadratic form statistic satisfying our assumptions (i)'' - (iv)'' as

$$\xi = [c R^{-1} c^T]^{1/(2m\delta)} \quad (1\text{-}62)$$

Notice that this reduces to the efficacy defined by (1-48) for the scalar case $L = 1$, and further to the efficacy defined by (1-41) when $m = 1$ and $\delta = 1/2$.

PROBLEMS

Problem 1.1

$X_1, X_2, ..., X_n$ are independent real-valued observations governed by the pdf

$$f(z | \theta) = \theta z^{\theta-1}, \quad 0 \le z \le 1$$

where the parameter θ has a value in $[1,\infty)$.

(a) Find the form of the best test for $H: \theta = \theta_0$ versus $K: \theta = \theta_1$. Show that the test can be interpreted as a comparison of the sum of a nonlinear function of each observation component to a threshold.

(b) Determine explicitly the best test of size $\alpha = 0.1$ for $\theta = 2$ versus $\theta = 3$. Is the test uniformly most powerful for testing $\theta = 2$ versus $\theta > 2$?

Problem 1.2

An observation X has pdf

$$f_X(z | \theta) = \frac{a}{2} e^{-a|z-\theta|}, \quad -\infty < z < \infty$$

where $a > 0$ is known. Consider size α tests, $0 < \alpha < 1$, for $\theta = 0$ versus $\theta > 0$ based on the single observation X.

(a) Show that a test which rejects $\theta = 0$ when $X > t$ is uniformly most powerful for $\theta > 0$.

(b) Show that a test which rejects $\theta = 0$ with probability γ when $X > 0$ is also a locally optimum test for $\theta > 0$.

(c) Sketch the power functions of these two tests. Verify that they have the same slope at $\theta = 0$.

Problem 1.3

$X = (X_1, X_2, ..., X_n)$ is a vector of independent and identically distributed observations, each governed by the pdf

$$f(x \mid \theta, a) = a e^{-a(x-\theta)}, \quad \theta < x < \infty$$

where θ and a are real-valued parameters. We want to test the simple null hypothesis $H: \theta = \theta_0, a = a_0$ against the simple alternative hypothesis $K: \theta = \theta_1 \leq \theta_0, a = a_1 > a_0$.

(a) Find in its simplest form the best test for H versus K. Let the alternative hypothesis K_1 be defined by $K_1: \theta \leq \theta_0, a > a_0$. Is your test uniformly most powerful for K_1?

(b) Let K_2 be the alternative hypothesis $\theta = \theta_0, a > a_0$. Is the above test uniformly most powerful for K_2?

(c) Obtain the generalized likelihood ratio test for H versus K_2, and compare it with your test found in (a).

Problem 1.4

$X = (X_1, X_2, ..., X_n)$ is a vector of independent and identically distributed components, each being Gaussian with mean μ and variance σ^2. For testing $H: \sigma^2 = 1$, μ unspecified, versus $K: \sigma^2 \neq 1$, μ unspecified, obtain the form of the *generalized likelihood ratio* test. Show that the test reduces to deciding if a particular test statistic falls inside some interval. Explain how the end points of the interval can be determined to obtain a specified value for the size α.

Problem 1.5

$X = (X_1, X_2, ..., X_n)$ is a vector of independent and identically distributed Gaussian components, each with mean θ and variance 1. Let $\bar{X}$ be the sample mean $\sum_{i=1}^{n} X_i/n$. For testing $\theta = 0$ versus $\theta \neq 0$, show that the test which rejects $\theta = 0$ when $|\bar{X}|$ exceeds a threshold is uniformly most powerful unbiased of its size. (Use Theorem 2.)

Problem 1.6

Given n independent and identically distributed observations X_i governed by a unit-variance Gaussian pdf with mean θ, a test for $\theta = 0$ versus $\theta > 0$ may be based on the test statistic $\sum_{i=1}^{n} X_i$.

Another possible test statistic for this problem is $\sum_{i=1}^{n} X_i^2$. Obtain the efficacies for these two tests, for suitable sequences of alternatives $\{\theta_n\}$ converging to 0, verifying that the regularity conditions (i) - (iv) or (i)' - (iv)' are satisfied. What can you say about the ARE of these two tests?

Problem 1.7

For the Cauchy density function

$$f(z \mid \theta) = \frac{1}{\pi} \frac{1}{1 + (z - \theta)^2}$$

with location parameter θ, a test for $\theta = 0$ versus $\theta > 0$ may be based on the *median* of n independent observations. Using results on the distributions of order statistics, obtain the efficacy of this test.

Problem 1.8

Independent observations $X_1, X_2, ..., X_n$ are governed by a common pdf f, which is unimodal and symmetric about its location value θ. The pdf f has a finite variance σ^2. Show that the ARE of the test for $\theta = 0$ based on the sample *median* relative to that based on the sample *mean* is $4\sigma^2 f^2(0)$. This result is independent of the scale parameter σ. Obtain its minimum value, and the corresponding unimodal symmetric pdf for which this minimum value of the ARE is achieved.

Chapter 2

DETECTION OF KNOWN SIGNALS IN ADDITIVE NOISE

2.1 Introduction

In this chapter we will begin our description of signal detection schemes for non-Gaussian noise by considering one of the simplest to formulate and most commonly encountered of all detection problems. This is the detection of a real-valued deterministic signal which is completely known in a background of additive noise. We shall be concerned exclusively with discrete-time detection problems in this book. In many applications discrete-time observations are obtained by sampling at a uniform rate some underlying continuous-time observation process. In the next section we will give the example of detection of a pulse train in additive noise to illustrate how a discrete-time detection problem may arise in a different manner from an original problem formulation in the continuous-time domain.

To obtain canonical results for the detector structures we will focus on the case of weak-signal detection when a large number of observation values are available, so that local expansions and asymptotic approximations can be used. We will then explain how the asymptotic performances of different detectors may be compared. In addition, we will discuss the value of such asymptotic performance comparison results for comparisons of performance in non-asymptotic cases of finite-length observation sequences and non-vanishing signal strengths.

Most of this development is for the detection problem in which the observations represent either noise only or a signal with additive noise. A common and important example of such a detection requirement is to be found in radar systems. In the communication context this detection requirement occurs in the case of on-off signaling. Furthermore, our development will be based primarily on the Neyman-Pearson approach, in which the performance criterion is based on the probability of correct detection for a fixed value of the probability of false detection or false alarm. In the penultimate section, however, we briefly discuss the case of binary signaling and the weak-signal version of the Bayes detector.

2.2 The Observation Model

For the known-signal-in-additive-noise detection problem we may describe our observation vector $X = (X_1, X_2, ..., X_n)$ of real-

valued components X_i by

$$X_i = \theta s_i + W_i, \quad i = 1,2,...,n \qquad (2\text{-}1)$$

Here the W_i are random noise components which are always present, and the s_i are the known values of the signal components. The signal amplitude θ is either zero (the observations contain no signal) or positive, and the detection problem is to decide on the basis of X whether we have $\theta = 0$ or we have $\theta > 0$.

2.2.1 Detection of a Pulse Train

As we have mentioned above, the X_i may be samples obtained from some continuous-time observation process. One such example occurs in the detection of a pulse train with a known pattern of amplitudes for the otherwise identical pulses in the train. For instance, the signal to be detected may be the bipolar pulse train depicted in Figure 2.1. In this case the X_i could be samples taken at the peak positions of the pulses (if present) from a noisy observed waveform. Alternatively, the X_i could be the outputs at specific time instants of some pre-detection processor, such as a filter matched to the basic pulse shape. It will be useful to consider a little more explicitly this mechanism for generating the observations X_i, before we continue with the model of (2-1).
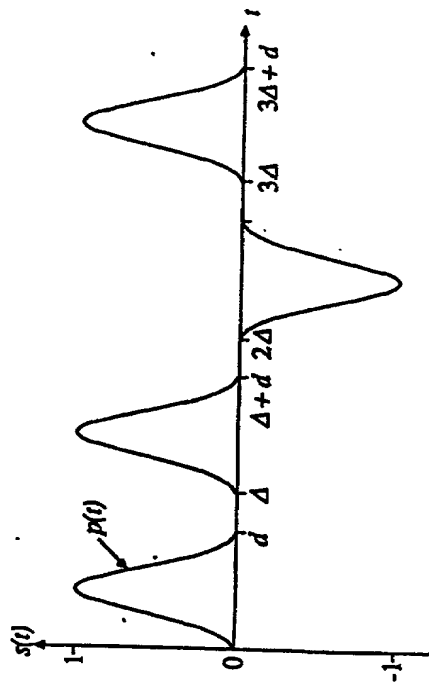

Figure 2.1 Bipolar Pulse Train

Let the $p(t)$ be the basic pulse shape in Figure 2.1, defined to be zero outside the interval $(0, \Delta)$. The train of n such non-overlapping pulses may be described by

$$s(t) = \sum_{i=1}^{n} c_i p(t - [i-1]\Delta) \qquad (2\text{-}2)$$

where c_i is the known amplitude of the i-th pulse. In Figure 2.1 the amplitudes c_i are all 1 or -1. The received continuous-time observation process $X(t)$ may be expressed as

$$X(t) = \theta s(t) + W(t), \quad T_0 \leq t \leq T_1 \qquad (2\text{-}3)$$

Now $W(t)$ is an additive continuous-time random noise process. In (2-3) the observation interval $[T_0, T_1]$ includes the signal interval $[0, n\Delta]$.

Let the noise process $W(t)$ have zero mean and let it be wide-sense stationary with power spectral density $\Phi_W(\omega)$. Let $P(\omega)$ be the Fourier transform of the pulse $p(t)$. Suppose we wanted to find that *linear* processor which maximizes its output signal-to-noise ratio (SNR) when the input is $X(t)$. The solution is the well-known *matched filter*, which is specified to have frequency response

$$\hat{H}_M(\omega) = \frac{\displaystyle\sum_{i=1}^{n} c_i P^*(\omega) e^{j\omega(i-1)\Delta}}{\Phi_W(\omega)} e^{-j\omega n\Delta} \qquad (2\text{-}4)$$

(Strictly speaking, this is the correct solution for long observation intervals.) The numerator above (without the delay term $e^{-j\omega n\Delta}$) is the Fourier transform of $s(-t)$. From the above we find that

$$\hat{H}_M(\omega) = \frac{P^*(\omega)}{\Phi_W(\omega)} e^{-j\omega n\Delta} \sum_{i=1}^{n} c_i e^{-j\omega(n - i)\Delta}$$
$$= H_M(\omega) \sum_{i=1}^{n} c_i e^{-j\omega(n - i)\Delta} \qquad (2\text{-}5)$$

where $H_M(\omega)$ is the frequency response of the filter matched for a *single pulse* $p(t)$, maximizing output SNR for this case at time $t = \Delta$.

Thus the maximum output SNR linear filter can be given the system interpretation of Figure 2.2. Here the output of the *single-pulse matched filter* is sampled at times $t = i\Delta$, $i = 1,2,...,n$, the i-th sampled value X_i is multiplied by c_i, and

the products summed together to form the final output. Because the input to the system is composed of additive signal and noise terms, the sampled values X_i consist of additive signal and noise components. We can denote the noise components as the W_i, and the signal components may be denoted as θc_i. From (2-3) we see that the c_i here are the outputs of the single-pulse matched filter at times $t = i\Delta$, when the input is the pulse train $s(t)$ of (2-2).
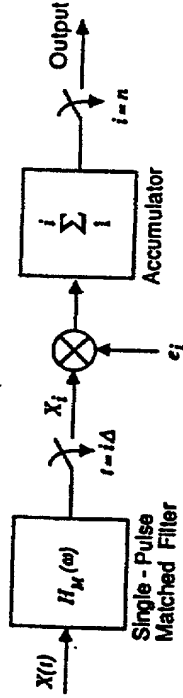
Figure 2.2 Linear System Maximizing Output SNR in Pulse Train Detection

If the noise process $W(t)$ in (2-3) is Gaussian, the system of Figure 2.2 leads to the *optimum* scheme for signal detection, that is, for testing the null hypothesis that $\theta = 0$ versus the alternative that $\theta > 0$. This is one of the central results of classical signal detection theory for Gaussian noise. The optimum detector uses the output of the linear system of Figure 2.2 as the test statistic, and compares it to a threshold to make a decision regarding the presence or absence of the signal.

The optimum detection scheme of Figure 2.2 can be given a particularly useful interpretation under some assumptions on the nature of the pulse train $s(t)$ and on the noise spectral density function $\Phi_W(\omega)$. Consider first the case where the noise is white, with a flat spectral density $\Phi_W(\omega) = N_0/2$. Then the impulse response $h_M(t)$ of the single-pulse matched filter is simply

$$h_M(t) = \frac{2}{N_0} p(\Delta - t) \qquad (2\text{-}6)$$

which is zero outside the interval $[0, \Delta]$, since $p(t)$ is confined entirely to this interval for non-overlapping pulses in $s(t)$. The covariance function $\rho_M(t)$ of the noise process at the output of the matched filter is then

$$\rho_M(t) = \frac{N_0}{2} h_M(t) * h_M(-t) \qquad (2\text{-}7)$$

which is therefore zero for $|t| \geq \Delta$. Thus the noise components W_i are uncorrelated, and hence independent for Gaussian noise.

More generally, the spectral density $\Phi_M(\omega)$ of the noise at the output of the single-pulse matched filter is

$$\Phi_M(\omega) = \Phi_W(\omega) \, |H_M(\omega)|^2$$
$$= P(\omega) \frac{P^*(\omega)}{\Phi_W(\omega)} \tag{2-8}$$

so that

$$\rho_M(t) = p(t) * h_M(t + \Delta) \tag{2-9}$$

Now the basic pulse $p(t)$ may in general occupy only a portion of the period Δ of the pulse train; let us therefore consider the case where $p(t)$ is non-zero only on the interval $[0, d]$, with $d \leq \Delta$. If the noise is not white the impulse response $h_M(t)$ of the single-pulse matched filter may have an effective duration which is larger than d. Let us now impose the condition, which will usually be satisfied for narrow pulses at least, that $h_M(t)$ is essentially zero outside of an interval of length $\Delta + (\Delta - d) = \Delta + \beta$. An examination of the functions shown in Figure 2.3 reveals that $\rho_M(t)$ will then be essentially zero for $|t| \geq \Delta$, so that we may again assume that the noise components W_i are uncorrelated and therefore independent for Gaussian noise. Under the same assumption the signal components e_i at the matched filter outputs will be simply

$$s_i = e_i \int_0^t p(t) h_M(\Delta - t) \, dt$$
$$= e_i \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{|P(\omega)|^2}{\Phi_W(\omega)} \, d\omega \tag{2-10}$$

We may assume without any loss of generality that the pulse $p(t)$ has been defined with a scale factor making the quantity multiplying e_i in (2-10) equal to unity. We can then describe the samples X_i as

$$X_i = \theta e_i + W_i \tag{2-11}$$

where the W_i are *independent and identically distributed* (i.i.d.) zero-mean Gaussian random variables.

The system of Figure 2.2 can be interpreted as being composed of two parts, a *single-pulse matched filter* generating the X_i and a *linear correlator* detector forming the test statistic by correlating the X_i with the e_i. Of course it is well-known that for X_i
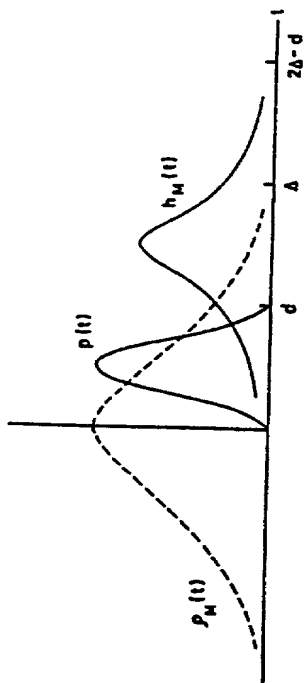
Figure 2.3 Filter Impulse Response and Output Autocovariance Function, of Matched Filter for Pulse Train Detection

given by (2-11) with i.i.d. Gaussian noise the linear-correlator detector is the uniformly most powerful (UMP) detector for $\theta > 0$.

What we have seen so far is that the linear processor maximizing the output SNR in detection of a pulse train may be considered quite generally to be operating on a set of intermediate observations X_i which are modeled by (2-1). Under a specific condition on the pulse and spectral density characteristics, the W_i in (2-1) are uncorrelated noise terms. If the noise process $W(t)$ is not Gaussian, the maximum output SNR scheme will not lead to an optimum detector. However, explicit solutions for the optimum detector are then also hard to come by. The use of a matched filter for SNR optimization is a common, simple, and generally well-founded engineering technique, and it usually makes good sense to continue to use it to generate the intermediate observations X_i even when the input noise is not Gaussian. However, as discussed at the end of Section 3.3 in Chapter 3, the use of linear predetection filtering is *not* appropriate when the noise process $W(t)$ contains impulsive non-Gaussian components. In the following development we will concentrate on the best use of the X_i modeled by (2-1) when the noise components W_i are not Gaussian.

2.2.2 Statistical Assumptions

Although the detection of a pulse train is only a particular example we chose to focus on, it illustrates the importance of the basic model of (2-1). We will now become more specific, and assume that the W_i form a sequence of i.i.d. random variables.

Their common univariate cumulative distribution function (cdf) will be denoted by F. Note that even though we do not require that F be Gaussian, we will assume that the W_i are independent. We will make some regularity assumptions about F that are generally met for all cases of interest. For future reference we will call these *Assumptions A*:

A. (i) F is absolutely continuous, so that a probability density function (pdf) f exists for the W_i;

(ii) f is absolutely continuous, so that its derivative $\frac{df(x)}{dx} = f'(x)$ exists for almost all x;

(iii) f' satisfies $\int_{-\infty}^{\infty} |f'(x)| \, dx < \infty$.

Assumptions A (i) and A (ii) are smoothness assumptions, and A (iii) is a technical requirement. These assumptions will allow us to justify certain mathematical operations, such as interchanges in the order of differentiation with respect to a parameter and integration of a function, although for the most part we will not provide details of such proofs.

This is also an appropriate place to introduce a second assumption, which we will call *Assumption B*:

B.
$$I(f) \triangleq \int_{-\infty}^{\infty} \left[\frac{f'(x)}{f(x)} \right]^2 f(x) \, dx \tag{2-12}$$
$$< \infty$$

The quantity $I(f)$ is called *Fisher's information* for location shift, and the above assumption says that f has *finite* Fisher's information for location shift. The significance of $I(f)$ will become clear later in this chapter. We will find that this assumption is also satisfied by most noise density functions of interest. Notice that Assumption B implies that $f'(X_i)/f(X_i)$ has a finite variance under the noise-only condition $\theta = 0$; its mean value then is zero.

In this book we are concerned with problems where the noise density function f is completely specified, as a special case of the general *parametric* problem where f may have a finite number of unknown parameters (such as the noise variance). Our detection problem can be formulated as a statistical hypothesis-testing problem of choosing between a *null hypothesis* H_1 and an *alternative hypothesis* K_1 describing the joint density function f_X of the

observation vector X, with

$$H_1: \ f_X(x) = \prod_{i=1}^{n} f(x_i) \tag{2-13}$$

$$K_1: \ f_X(x) = \prod_{i=1}^{n} f(x_i - \theta s_i), \ s \text{ specified, any } \theta > 0 \tag{2-14}$$

Here s is the vector $(s_1, s_2, \ldots, s_n)$ of signal components. Note that we are considering parametric hypotheses which completely define f_X to within a finite number of unknown parameters (here with only $\theta > 0$ unknown under the alternative hypothesis). Let us now proceed to obtain the structures of tests for H_1 versus K_1.

2.3 Locally Optimum Detection and Asymptotic Optimality

Since the alternative hypothesis K_1 is not a simple hypothesis, the signal amplitude value being unspecified, we cannot apply directly the fundamental lemma of Neyman and Pearson to obtain the structure of the optimum detector for our detection problem. For non-Gaussian noise densities it is also generally impossible to obtain UMP tests for the composite alternative hypothesis K_1.

To illustrate the difficulty, consider the special case where f is specified to be the *double-exponential* noise density function defined by

$$f(x) = \frac{a}{2} e^{-a|x|}, \ a > 0 \tag{2-15}$$

The likelihood ratio for testing H_1 versus K_1, for a particular value $\theta = \theta_0 > 0$, is

$$L(X) = \prod_{i=1}^{n} \frac{f(X_i - \theta_0 s_i)}{f(X_i)} \tag{2-16}$$

This now becomes

$$L(X) = e^{-a \sum_{i=1}^{n}(|X_i - \theta_0 s_i| - |X_i|)} \tag{2-17}$$

giving

$$\ln L(X) = a \sum_{i=1}^{n}(|X_i| - |X_i - \theta_0 s_i|) \tag{2-18}$$

Thus for *given* $\theta = \theta_0$, the test based on

$$\hat{\lambda}(X) = \sum_{i=1}^{n} (|X_i| - |X_i - \theta_{0si}|) \qquad (2\text{-}19)$$

is an optimum test, since the constant s is positive. The optimum detector therefore has a test function defined by

$$\delta(X) = \begin{cases} 1, & \hat{\lambda}(X) > t \\ r, & \hat{\lambda}(X) = t \\ 0, & \hat{\lambda}(X) < t \end{cases} \qquad (2\text{-}20)$$

where the threshold t and randomization probability r are chosen to obtain the desired value for the false-alarm probability p_f, so that the equation

$$E\{\delta(X) | H_1\} = p_f \qquad (2\text{-}21)$$

is satisfied. Notice that we do not need randomization at $\hat{\lambda}(X) = t$ if this event has zero probability under H_1.

We can express $\hat{\lambda}(X)$ of (2-19) in the form

$$\hat{\lambda}(X) = \sum_{i=1}^{n} l(X_i; \theta_{0si}) \qquad (2\text{-}22)$$

where the characteristic l is defined by

$$l(x; \theta_s) = |x| - |x - \theta_s| \qquad (2\text{-}23)$$

This is shown in Figure 2.4 as a function of x and depends strongly on θ, so that $\hat{\lambda}(X)$ cannot be expressed in a simpler form decoupling θ_0 and the X_i. For an implementation of the test statistic $\hat{\lambda}(X)$ the value θ_0 of θ must be known, and a UMP test does not exist for this problem for $n > 1$.

One approach we might take in the above case is to use a *generalized likelihood ratio (GLR)* test, here obtained by using as the test statistic $\hat{\lambda}(X)$ of (2-19) with θ_0 replaced by its *maximum likelihood (ML)* estimate under the alternative hypothesis K_1. This maximum-likelihood estimate $\hat{\theta}_{ML}$ is given implicitly as the solution of the equation

$$\sum_{i=1}^{n} s_i \, \text{sgn}(X_i - \hat{\theta}_{ML} s_i) = 0 \qquad (2\text{-}24)$$

where
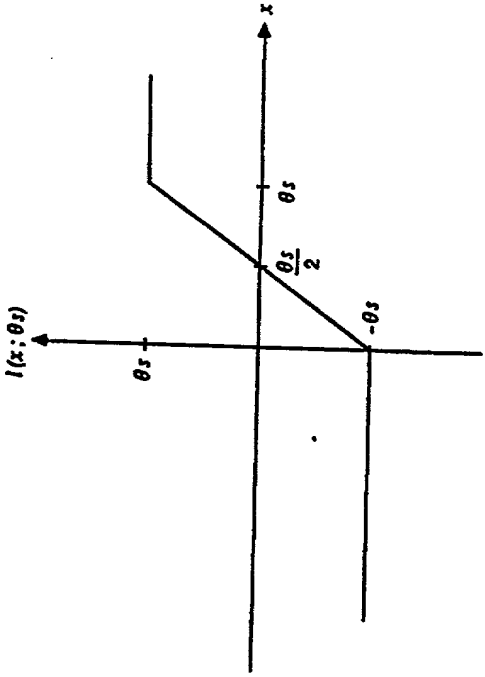
Figure 2.4 The Characteristic $l(x; \theta_s)$ of Equation (2-23)

$$\text{sgn}(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases} \qquad (2\text{-}25)$$

provided that the solution turns out to be non-negative; otherwise, $\hat{\theta}_{ML} = 0$ (Problem 2.1). Thus the implementation of the GLR test is not simple. In addition, the distribution of the GLR test statistic under the null hypothesis is not easily obtained.

In the general case, for any noise density function f, the optimum detector for *given* $\theta = \theta_0 > 0$ under K_1 can be based on the test statistic

$$\lambda(X) = \ln L(X)$$

$$= \sum_{i=1}^{n} \ln \frac{f(X_i - \theta_0 s_i)}{f(X_i)} \qquad (2\text{-}26)$$

which is of the form of $\hat{\lambda}(X)$ of (2-22). But again, θ_0 must be specified and the detector will be optimum only for a signal with that amplitude. The GLR detector can be obtained if the ML estimate $\hat{\theta}_{ML}$ of θ can be found under the constraint that $\hat{\theta}_{ML}$ be non-negative. Once again, in general this will not lead to an easily implemented and easily analyzed system.

$$\frac{d}{d\theta} P_x(\theta|D)\Big|_{\theta=0} = \int_x \delta(x) \frac{d}{d\theta} \prod_{i=1}^{n} f(x_i - \theta c_i)\Big|_{\theta=0} dx$$

$$= \int_x \delta(x) \left[\sum_{i=1}^{n} -c_i \frac{f'(x_i)}{f(x_i)}\right] \prod_{i=1}^{n} f(x_i)\, dx$$

$$= E\left\{\delta(X) \left[\sum_{i=1}^{n} -c_i \frac{f'(X_i)}{f(X_i)}\right] \Big| H_1\right\} \quad (2-31)$$

From this it follows, from the generalized Neyman-Pearson lemma, that a locally optimum detector D_{LO} is based on the test statistic

$$\lambda_{LO}(X) = -\sum_{i=1}^{n} c_i \frac{f'(X_i)}{f(X_i)}$$

$$= \sum_{i=1}^{n} c_i\, g_{LO}(X_i) \quad (2-32)$$

where g_{LO} is the function defined by

$$g_{LO}(z) = \frac{-f'(z)}{f(z)} \quad (2-33)$$

Note that we may express $\lambda_{LO}(X)$ as

$$\lambda_{LO}(X) = \sum_{i=1}^{n} c_i \frac{d}{d\theta} \ln f(X_i - \theta c_i)\Big|_{\theta=0}$$

$$= \frac{d}{d\theta} \sum_{i=1}^{n} \ln \frac{f(X_i - \theta c_i)}{f(X_i)}\Big|_{\theta=0} \quad (2-34)$$

from which the LO detector test statistic (multiplied by θ) is seen to be a first-order approximation of the optimum detector test statistic given by (2-26).

The development of detection schemes for weak-signal situations can be traced back to the works of Middleton [1960] and Capon [1961]. Middleton used an expansion of the likelihood ratio in a Taylor series, from which an LO detection statistic is obtained as a weak-signal approximation. Capon considered explicitly the property of the LO detector as an optimum detector maximizing the slope of the power function for vanishing signal strength. An elaboration of Middleton's initial work is given in [Middleton, 1966]. Other early works employing the idea of a series expansion of the likelihood-ratio are those of Rudnick [1961],

Algazi and Lerner [1964], and Antonov [1967a, 1967b]. Helstrom [1967, 1968] also briefly discusses "threshold" or LO detection.

2.3.2 Generalized Correlator Detectors

The LO detector test statistic is, of course, not dependent on θ and has an easily implemented form. Let us define a *generalized correlator* (GC) test statistic $T_{GC}(X)$ as a test statistic of the form

$$T_{GC}(X) = \sum_{i=1}^{n} c_i\, g(X_i) \quad (2-35)$$

where the c_i, $i=1,2,...,n$ are a set of constants which are correlated with the $g(X_i)$, $i=1,2,...,n$ to form $T_{GC}(X)$. The characteristic g is a memoryless or instantaneous nonlinearity. Then it is clear that $\lambda_{LO}(X)$ is a GC test statistic for which g is the locally optimum nonlinearity g_{LO} and the coefficients c_i are the known signal components θ_i. Figure 2.6 shows the structure of a GC detector, from which it is clear that its implementation is quite easy. To set the threshold for any desired false-alarm probability P_f, the distribution of $T_{GC}(X)$ under the null hypothesis is required. The fact that $T_{GC}(X)$ is a linear combination of the i.i.d. $g(X_i)$ under the null hypothesis leads to some simplification of the problem of threshold computation. In practice the null-hypothesis distribution of $T_{GC}(X)$ may be computed through numerical convolution of the density functions of the $c_i g(X_i)$. If n is large enough, the Gaussian approximation may be used for the distribution of $T_{GC}(X)$, as we will see later.
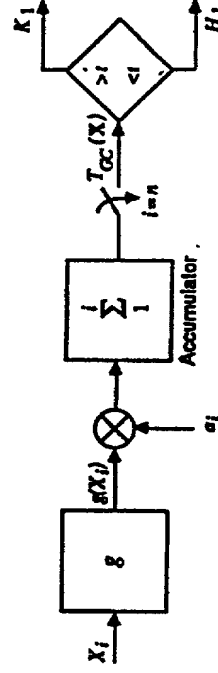
Figure 2.6 Generalized Correlator (GC) Detector

For f a zero-mean Gaussian density with variance σ^2, we have

$$g_{LO}(z) = \frac{z}{\sigma^2} \quad (2-36)$$

so that the locally optimum detector is based on a *linear correlator (LC)* test statistic

$$T_{LC}(X) = \sum_{i=1}^{n} c_i X_i \qquad (2\text{-}37)$$

This is obtained from $\lambda_{LO}(X)$ of (2-32) using (2-36), after dropping the constant σ^2. This linear correlator detector based on $T_{LC}(X)$ is known to be not just locally optimum but also UMP for $\theta > 0$ when the noise is Gaussian. Clearly the LC test statistic is a special case of a GC test statistic.

For the double-exponential noise density of (2-15) we find that g_{LO} is given by

$$g_{LO}(z) = a \, \text{sgn}(z) \qquad (2\text{-}38)$$

Note that the *optimum* detector for $\theta = \theta_0$ in this case is based on the test statistic $\tilde{\lambda}(X)$ of (2-19). An equivalent detector is one based on $a\lambda(X)/\theta_0$ (since θ_0 and a are assumed known), which converges to $\lambda_{LO}(X)$ of (2-32) as $\theta_0 \to 0$. The LO detector for the double-exponential density is therefore a *sign correlator (SC)* detector based on the GC statistic

$$T_{SC}(X) = \sum_{i=1}^{n} c_i \, \text{sgn}(x_i) \qquad (2\text{-}39)$$

In the case of constant signals we may set the c_i to be unity; the result is what is called the *sign detector*, of which the SC detector is a general form. (We will study the sign and SC detectors in detail as *nonparametric* detectors in a sequel to this book.) Their property of being robust detectors is also well known.

The LC and the SC detectors (and their special cases for $c_i = 1$, all i, the *linear* and *sign* detectors, respectively) are the best-known of the LO detectors for known signals in additive i.i.d. noise. We will, in fact, encounter other versions of these detectors in Chapters 5, 6, and 7. The LC detector is optimum for the important special case of Gaussian noise, and therefore is used as a standard against which other detectors are compared for performance under Gaussian noise. The SC detector is a particularly simple example of a detector which is optimum for a non-Gaussian noise density, which does happen to be a useful simple model of non-Gaussian noise in several applications. Its very simple structure together with its nonparametric and robust performance characteristics make it, too, a commonly used standard against which other detectors are compared for performance and complexity.

2.3.3 Asymptotic Optimality

For our detection problem formalized as that of testing H_1 versus K_1, a locally optimum detector D_{LO} of size α has a power function $p_d(\theta | D_{LO})$ for which the slope $p_d'(\theta | D_{LO})$ at $\theta = 0$ is a maximum from among all detectors D of size α. In the non-Gaussian situation an LO detector will generally not be a UMP detector. For some *particular* value $\theta_0 > 0$ for θ we can find the *optimum* (Neyman-Pearson) detector D_{NP} of size α; let its power function be explicitly written as $p_d(\theta | \theta_0, D_{NP})$. Then in general we will have

$$p_d(\theta_0 | D_{LO}) < p_d(\theta_0 | \theta_0, D_{NP}) \qquad (2\text{-}40)$$

for $0 < \alpha < 1$, and this will be true for θ_0 arbitrarily small.

Let us call the function of θ_0 on the right-hand side of (2-40) the *envelope power function*, which is therefore the function $p_d(\theta | E)$ defined by

$$p_d(\theta | E) = p_d(\theta | \theta, D_{NP}) \qquad (2\text{-}41)$$

Clearly, we will find that $\lim_{\theta \to 0} p_d(\theta | E) = \alpha$. Since we have assumed the non-existence of a UMP detector, no *single* detector exists which has the power function $p_d(\theta | E)$. Rather, the envelope power function now becomes the standard against which the performances of other detectors may be compared. The relationship between these power functions is illustrated in Figure 2.5.

In using an LO detector we can be assured only that $p_d'(0 | D)$ is maximized. In justifying the use of an LO detector we have mentioned that the case of weak signals implied by the condition $\theta \to 0$ is an important case in practical applications. On the other hand note that for any *fixed* sample size n, the power or detection probability approaches the value α (the false-alarm probability) for $\theta \to 0$. Thus while the condition $\theta \to 0$ is of considerable interest, it is generally used *in conjunction with* an assumption that $n \to \infty$. The practical significance is that in detection of a weak signal one necessarily uses a relatively large sample size to get a reasonable value for the probability of correct detection. We see that with local optimality we addressed only one part (the "local signal" case) of this combined asymptotic situation. We will now consider the combined asymptotic assumption $\theta \to 0$ and $n \to \infty$, and show that in general the LO detector has an optimality property for this asymptotic case.

The problem can be formally stated as a sequence of hypothesis-testing problems, $\{H_{1,n}$ versus $K_{1,n}, n = 1,2,...\}$ where $H_{1,n}$ is simply an explication of the fact that H_1 of (2-13) is applicable for the case of n independent observations from the noise density. The sequence of alternatives $\{K_{1,n}, n = 1,2,...\}$ is described by

$$K_{1,n}: \ f_X(x) = \prod_{i=1}^{n} f(x_i - \theta_n s_i), \quad n = 1,2,...,$$

$$\theta_n > 0, \ \theta_n \to 0 \ . \qquad (2\text{-}42)$$

We shall be interested in particular types of sequences $\{\theta_n\}$ converging to zero. For example, we could have $\theta_n = \gamma/\sqrt{n}$ for some fixed $\gamma > 0$, so that $n\theta_n^2$ remains fixed at γ^2 as $n \to \infty$. In the *constant signal case* ($s_i = 1$, all n) this models a sequence of detection problems with increasing sample sizes and decreasing signal amplitudes in such a way that the *total signal energy* $\theta_n^2 \sum_{i=1}^{n} s_i^2 = n\theta_n^2$ remains fixed. In the *time-varying signal case*, the condition that $\frac{1}{n} \sum_{i=1}^{n} s_i^2$ converges (without loss of generality, to unity) as $n \to \infty$ allows a similar interpretation to be made in the asymptotic case. Let us, in fact, henceforth make such an assumption about our known signal sequence $\{s_i, i = 1,2,...\}$, and also require it to be uniformly bounded for mathematical convenience. These may be stated as *Assumptions C*:

C. (i) There exists a finite, non-zero bound U, such that

$$0 \leq |s_i| \leq U, \ , \quad i = 1,2,... \qquad (2\text{-}43)$$

(ii) The asymptotic average signal power is finite and non-zero,

$$0 < \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} s_i^2 = P_s^2 < \infty \qquad (2\text{-}44)$$

We should note one more technical detail. In defining the $K_{1,n}$ by (2-42), there is the implication that as n increases the known-signal vector $s = (s_1, s_2, ..., s_n)$ grows by having *additional* components appended to it, without change in the *existing* components. This appropriately models a situation where the observations X_i are obtained as samples taken at a *fixed* sampling rate from some continuous-time waveform, with the observation time *increasing* to generate the sequence of hypothesis-testing problems. On the other hand, one may have a situation where the observation interval is *fixed*, with the sampling rate *increasing* to generate the sequence of hypothesis-testing problems. Such cases are handled by replacing the s_i in (2-42) with $s_{i,n}$, thus explicitly allowing the set of n signal components under $K_{1,n}$ to depend more generally on n.

The idea in considering formally a *sequence* of hypothesis-testing problems $\{H_{1,n}$ versus $K_{1,n}, n = 1,2,...\}$ is the following: if we can find a corresponding sequence $\{D_n, n = 1,2,...\}$ of detectors which in the limit $n \to \infty$ have some optimality property, then for sufficiently large finite n the use of D_n will give approximately optimum performance.

Since we are considering a sequence of hypothesis-testing problems indexed by n, we will now modify our notation slightly and use, for instance, $p_d(\theta_n | D_n)$ for the power function of a size-α detector D_n based on n observations $X_1, X_2, ..., X_n$ for $H_{1,n}$ versus $K_{1,n}$. What we would like to do is consider sequences $\{\theta_n\}_{n=1}^{\infty}$ converging to zero in such a way that the envelope power function values $p_d(\theta_n | E_n)$ converge to values which lie strictly between α and 1 (for size-α detectors). Then we would be looking at a sequence of detection problems in which the sample size is growing and signal amplitude is shrinking in such a way as to allow the optimum detectors always to yield a useful level of asymptotic detection performance (detection probability larger than α) but without allowing asymptotically "perfect" detection (detection probability less than unity).

Definition: We will say that a sequence of detectors $\{D_{AO,n}, n = 1,2,...\}$ is *asymptotically optimum* (AO) at level α for $\{H_{1,n}$ versus $K_{1,n}, n = 1,2,...\}$ if[1]

$$(i) \qquad \lim_{n \to \infty} E\{\delta_{AO,n}(X)|H_{1,n}\}$$

$$= \lim_{n \to \infty} p_d(0|D_{AO,n}) \leq \alpha \qquad (2\text{-}45)$$

and

$$(ii) \qquad \lim_{n \to \infty} [p_d(\theta_n | E_n) - p_d(\theta_n | D_{AO,n})] = 0 \qquad (2\text{-}46)$$

Here $\delta_{AO,n}(X)$ is the test function for the detector $D_{AO,n}$.

Condition (i) above will obviously be satisfied if each detector $D_{AO,n}$ in the sequence is a size-α detector. According to condition (ii), an AO sequence of detectors has a sequence of power values for the alternatives $K_{1,n}$ defined by the θ_n, which is in the *limit* $n \to \infty$ equal to the power of the optimum detector. If

---

[1] (i) May be made more general by replacing " $\lim_{n \to \infty}$ " by "lim sup "; this would not necessarily require $p_d(0|D_{AO,n})$ to converge as $n \to \infty$.

θ_n, $n = 1,2,...$, defines a sequence of alternatives for which the sequence of optimum size-α detectors is *consistent*, then
$$\lim_{n \to \infty} P_I(\theta_n | E_n) = 1.$$
In this case any other sequence of detectors which is simply consistent (and has asymptotic size α) will be an AO sequence. Thus it will be of most interest to consider cases for which $\alpha < \lim_{n \to \infty} P_I(\theta_n | E_n) < 1$.

Note carefully that the property of being AO belongs to a *sequence* of detectors. Usually, however, we consider sequences of detectors employing exactly the same *rule* for computing the test statistic for each n; for example, we might consider a sequence of SC detectors with test statistics defined by (2-39) for each n. In such cases we often say that a particular detector is AO, when we really mean that the sequence of such detectors is AO. The property of being locally optimum, on the other hand, belongs to an *individual* detector (operating on some sample of fixed size n).

2.3.4 Asymptotic Optimality of LO Detectors

Let us consider the sequence $\{D_{LO,n}, n = 1,2,...\}$ of LO detectors of size α for our sequence of hypothesis-testing problems. Then we can show that this is an AO sequence of detectors for the alternatives $K_{1,n}$ defined by $\theta_n = \gamma/\sqrt{n}$, $n = 1,2,...$, for any fixed $\gamma > 0$. This optimality property of the sequence of LO detectors gives them a stronger justification for use when sample sizes are large. To prove the asymptotic optimality of any sequence of detectors $\{D_n\}$ we have to be able to obtain the limiting values of the powers $p_I(\theta_n | D_n)$ and $p_I(\theta_n | E_n)$, and these can be obtained if the asymptotic distributions of the test statistics are known. For the envelope power function $p_I(\theta_n | E_n)$ we have to consider the optimum detectors based on the log-likelihood functions $\lambda(X)$ of (2-26), and obtain their asymptotic distributions for $\theta_0 = \theta_n$ and $n \to \infty$. Let us now carry out this asymptotic analysis in a heuristic way, with any regularity conditions required to make our analysis rigorously valid implicitly assumed to hold.

The log-likelihood function $\lambda(X)$ of (2-26) with $\theta_0 = \gamma/\sqrt{n}$ may be expanded in the Taylor series

$$\lambda(X) = \sum_{i=1}^n \ln \frac{f(X_i)}{f(X_i)} + \frac{\gamma}{\sqrt{n}} \sum_{i=1}^n s_i \left[\frac{-f'(X_i)}{f(X_i)} \right]$$
$$+ \frac{1}{2} \frac{\gamma^2}{n} \sum_{i=1}^n s_i^2 \left\{ \frac{f''(X_i)}{f(X_i)} - \left[\frac{f'(X_i)}{f(X_i)} \right]^2 \right\}$$
$$+ o(1)$$
$$\approx \frac{\gamma}{\sqrt{n}} \lambda_{LO}(X)$$

$$+ \frac{1}{2} \frac{\gamma^2}{n} \sum_{i=1}^n s_i^2 \left\{ \frac{f''(X_i)}{f(X_i)} - \left[\frac{f'(X_i)}{f(X_i)} \right]^2 \right\}$$ (2-47)

for large n. Note that a quantity Z_n is said to be of order $o(1/a_n)$ if $a_n Z_n \to 0$ (in probability). Thus the only essential way in which the above approximation differs from the LO test statistic $\lambda_{LO}(X)$ of (2-32) is in the second term above. But this term converges in probability,

$$\lim_{n \to \infty} \frac{\gamma^2}{2n} \sum_{i=1}^n s_i^2 \left\{ \frac{f''(X_i)}{f(X_i)} - \left[\frac{f'(X_i)}{f(X_i)} \right]^2 \right\}$$
$$= \frac{1}{2} \gamma^2 P_s^2 E \left\{ \frac{f''(X_i)}{f(X_i)} - \left[\frac{f'(X_i)}{f(X_i)} \right]^2 \right\}$$
$$= -\frac{1}{2} \gamma^2 P_s^2 I(f)$$ (2-48)

under both $H_{1,n}$ and $K_{1,n}$. We have assumed that $\int_{-\infty}^{\infty} f''(x)\,dx$ is zero. Note that P_s^2 is the average known-signal power and $I(f)$ is Fisher's information for location shift of (2-12). The first term $(\gamma/\sqrt{n})\lambda_{LO}(X)$ is asymptotically Gaussian under $H_{1,n}$, with mean value 0 and variance $\gamma^2 P_s^2 I(f)$. To find the asymptotic mean and variance of $(\gamma/\sqrt{n})\lambda_{LO}(X)$ under $K_{1,n}$, we expand it as

$$\frac{\gamma}{\sqrt{n}} \lambda_{LO}(X) = \frac{\gamma}{\sqrt{n}} \sum_{i=1}^n s_i \left[\frac{-f'(W_i + \gamma/\sqrt{n}\ s_i)}{f(W_i + \gamma/\sqrt{n}\ s_i)} \right]$$
$$= \frac{\gamma}{\sqrt{n}} \sum_{i=1}^n s_i \left[\frac{-f'(W_i)}{f(W_i)} \right]$$
$$+ \frac{\gamma^2}{n} \sum_{i=1}^n s_i^2 \left\{ \frac{-f''(W_i)}{f(W_i)} + \left[\frac{f'(W_i)}{f(W_i)} \right]^2 \right\}$$
$$+ o(1)$$ (2-49)

We conclude that the asymptotic mean of $(\gamma/\sqrt{n})\lambda_{LO}(X)$ is $\gamma^2 P_s^2 I(f)$ and its asymptotic variance is also $\gamma^2 P_s^2 I(f)$ under $K_{1,n}$. Furthermore, $(\gamma/\sqrt{n})\lambda_{LO}(X)$ is also asymptotically normally distributed under the sequence of alternatives $K_{1,n}$.

To summarize, then, we have that the sequence of optimum detection test statistics converges in distribution to a test statistic with a Gaussian distribution with mean $-\frac{1}{2}\gamma^2 P_s^2 I(f)$ and vari-

ance $\gamma^2 P_e^2 I(f)$ under the null hypotheses, and to a test statistic with a Gaussian distribution with mean $\frac{1}{2}\gamma^2 P_e^2 I(f)$ and variance $\gamma^2 P_e^2 I(f)$ under the alternative hypotheses. The sequence of LO test statistics $(\gamma/\sqrt{n})\lambda_{LO}(x)$ is similarly asymptotically Gaussian, with mean zero and variance $\gamma^2 P_e^2 I(f)$ under the null hypotheses and with mean and variance both equal to $\gamma^2 P_e^2 I(f)$ under the alternative hypotheses. From this we conclude that for the optimum detectors we get, with Φ the standard Gaussian distribution function,

$$\lim_{n\to\infty} P_d\left(\frac{\gamma}{\sqrt{n}}\mid E_n\right) = \lim_{n\to\infty} P\{\lambda(X) > t_n \mid K_{1,n}\}$$

$$= 1 - \Phi\left[\frac{t_n - \frac{1}{2}\gamma^2 P_e^2 I(f)}{\gamma P_e \sqrt{I(f)}}\right] \quad (2\text{-}50)$$

where the thresholds t_n for size-α optimum detectors converge to t_α given by

$$\alpha = 1 - \Phi\left[\frac{t_\alpha + \frac{1}{2}\gamma^2 P_e^2 I(f)}{\gamma P_e \sqrt{I(f)}}\right] \quad (2\text{-}51)$$

This gives

$$\lim_{n\to\infty} P_d\left(\frac{\gamma}{\sqrt{n}}\mid D_{LO,n}\right) = 1 - \Phi[\Phi^{-1}(1-\alpha) - \gamma P_e \sqrt{I(f)}] \quad (2\text{-}52)$$

Similarly, the sequence of LO detectors $D_{LO,n}$ using $(\gamma/\sqrt{n})\frac{1}{2}\lambda_{LO}(X)$ as test statistics and a fixed threshold $t_\alpha + \frac{1}{2}\gamma^2 P_e^2 I(f)$ have asymptotic size α and asymptotic power

$$\lim_{n\to\infty} P_d\left(\frac{\gamma}{\sqrt{n}}\mid D_{LO,n}\right)$$

$$= \lim_{n\to\infty} P\left\{\frac{\gamma}{\sqrt{n}}\lambda_{LO}(X) > t_\alpha + \frac{1}{2}\gamma^2 P_e^2 I(f)\mid K_{1,n}\right\}$$

$$= 1 - \Phi\left[\frac{t_\alpha - \frac{1}{2}\gamma^2 P_e^2 I(f)}{\gamma P_e \sqrt{I(f)}}\right]$$

$$= \lim_{n\to\infty} P_d\left(\frac{\gamma}{\sqrt{n}}\mid E_n\right) \quad (2\text{-}53)$$

This makes the sequence $\{D_{LO,n}, n = 1,2,...\}$ an AO sequence of

detectors for $\theta_n = \gamma/\sqrt{n}$ under $K_{1,n}$.

The above development does not constitute a rigorous proof of the fact that the sequence of LO detectors are AO for the alternatives $\theta_n = \gamma/\sqrt{n}$. A rigorous proof along the above lines would require additional specific regularity conditions to be imposed on the noise density functions f. Actually, it is possible to establish quite rigorously the required asymptotic normality results above without assuming regularity conditions beyond those of assumptions A, B, and C. It would not be appropriate to enter into the details of such a proof here, involving as it does some lengthy mathematical details. We shall observe here only that such a proof makes use of some very useful results known as LeCam's lemmas, which are given detailed exposure by Hajek and Sidak [1967, Ch. VI].

The sequence of LO detectors is not the only sequence of detectors which is AO for $\theta_n = \gamma/\sqrt{n}$. The sequence of optimum detectors is an obvious example of another AO sequence of detectors. Loosely speaking, any sequence of detectors with the correct asymptotic size and with test statistics which converge to LO test statistics as $n\to\infty$ will be an AO sequence of detectors. For example, consider the sequence of test statistics

$$T(X) = \sum_{i=1}^{n} c_i q_n(X_i), \quad n = 1,2,... \quad (2\text{-}54)$$

where

$$q_n(x) = \begin{cases} 1, & x > \frac{1}{n} \\ 0, & -\frac{1}{n} \le x \le \frac{1}{n} \\ -1, & x < -\frac{1}{n} \end{cases} \quad (2\text{-}55)$$

Then it can be shown easily that the sequence of size-α detectors based on this sequence of test statistics is AO for $\theta_n = \gamma/\sqrt{n}$ and for the double-exponential noise density. The LO detector for any specific value of the sample size may not be very convenient to implement. If we use instead a simpler detector for each n, defined in such a way that this sequence of detectors is asymptotically optimum, then for large n and small θ we will get performance very close to that of the LO detector.

We have given a careful and somewhat lengthy discussion of AO detectors in this section for several reasons. First, there has been some confusion in earlier work with regard to the asymptotic optimality properties of LO detectors. Second, we will find that this careful discussion will help us to better understand the

asymptotic relative efficiency (ARE) as a relative performance measure for two (sequences of) detectors. Finally, having given this development of asymptotic optimality specifically for the known signal detection problem, we will feel justified in omitting its treatment for other types of detection problems. It is possible to obtain AO detectors for many other types of detection problems, including those formulated for continuous-time observations.

We have already referred to the book by Hajek and Sidak [1967] as an excellent, though somewhat advanced treatment of this topic. In signal detection applications the asymptotic optimality criterion has been discussed by Levin and Kushnir [1969], by Levin and Rybin [1969], by Levin, Kushnir, and Pinskiy [1971], and by Kutoyants [1975, 1976], among others. One topic we have omitted is that of AO detection in *correlated* noise, which would have taken us too far outside the intended scope of this book. The interested reader is referred to the works of Pinskiy [1971], Poor [1982], Poor and Thomas [1979, 1980], and Halverson and Wise [1980a, 1980b] as examples of such investigations.

2.4 Detector Performance Comparisons

We have noted that the linear correlator detector based on the test statistic $T_{LC}(X)$ of (2-37) is a UMP detector for the known-signal detection problem when the noise is Gaussian. When the noise is not Gaussian, however, the LO detector is a generalized correlator detector using test statistic $\lambda_{LO}(X)$ of (2-32), a special of $T_{GC}(X)$ of (2-35). Since the use of detectors which are optimum under Gaussian conditions is widespread, we will be particularly interested now in comparing the performance of the LC detector with other GC detectors for different noise density functions f, for testing H_1 versus K_1. For example, it would be useful to know how well the LC detector performs relative to the sign correlator (SC) detector when the noise has the double-exponential density function, and also how much better it is than the SC detector when the noise is, indeed, Gaussian.

In comparing the performances of two detectors D_A and D_B in testing H_1 versus K_1 the relevant quantities which have to be considered are the sample size n, the signal amplitude θ, the false-alarm probability p_f, the noise density function f, and the detection probabilities $p_d(\theta|D_A)$ and $p_d(\theta|D_B)$. For a given density function f, for example, one can look at different combinations of n and θ and obtain for each the relationship between p_f and the detection probability p_d, the *receiver operating characteristics* (ROC), for each detector. Alternatively, one may obtain for each detector the power functions $p_d(\theta|D)$ for fixed f and different combinations of the sample size n and false-alarm probability $p_f = \alpha$.

It can be appreciated that this can become a rather formidable task even for simple GC detectors. Consider, for example, the LC detector when the noise density function is not Gaussian. To obtain its power function for a given n and size α one has generally to resort to a numerical technique to find the distribution of the LC test statistic for the given n. The comparison is even more difficult for the GC test statistics using nonlinear functions g.

Even if it were easy to generate power functions or ROCs for the detectors being compared, there would be entire families of such performance functions, parameterized by n and α or n and θ, respectively, for each detector for any noise density function f. What would be very useful would be to have a real-valued relative performance index which can be derived as a function of f, summarizing some particularly relevant aspect of the exhaustive performance comparison. These considerations lead us to the introduction of the asymptotic relative efficiency (ARE) of two detectors as a measure of their relative performance.

2.4.1 Asymptotic Relative Efficiency and Efficacy

In Section 1.6 of Chapter 1 we gave the technical details about the ARE as a measure of the relative performance of two detectors. We now explain it and illustrate its use specifically for the known-signal detectors. The most important thing to note about the ARE is that it measures the relative performance of two detectors in the *asymptotic* case $n \to \infty$ for a *sequence* of hypothesis-testing problems. In the specific context of known-signal detection we consider the sequence $\{H_{1,n}$ versus $K_{1,n}, n = 1,2,...\}$. We have $\theta_n \to 0$ as $n \to \infty$, and θ_n is chosen in a manner which makes the test statistics of both detectors have well-defined asymptotic distributions (of the same type) as $n \to \infty$. If under $H_{1,n}$ the two sequences of test statistics also have well-defined asymptotic distributions of a common type, the *parameters* of these four asymptotic distributions may be used to characterize the asymptotic relative performance of the two detectors. Under the specific assumptions stated in Section 1.6 (asymptotic distributions are Gaussian, each detector has test statistics which have the same asymptotic variance under both $H_{1,n}$ and $K_{1,n}$), the asymptotic means and variances can be used to define the ARE as an index of relative performance. The ARE has a numerical value which can be obtained for given f.

Let us now get down to specifics for GC detectors. Consider in general the use of a GC detector using coefficients a_{i1} as described by the test statistic $T_{GC}(X)$ of (2-35). Since we will be considering a sequence of such GC detectors based on some fixed g, in the limiting case there will be an infinite number of

coefficients a_i defining $T_{GC}(X)$. We make the following regularity Assumptions D:

D. (i) There exists a finite non-zero bound U_a such that

$$0 \leq |a_i| \leq U_a, \quad i = 1, 2, \ldots \quad (2\text{-}56)$$

(ii) The asymptotic average coefficient power is finite and non-zero,

$$0 < \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} a_i^2 = P_a^2 < \infty \quad (2\text{-}57)$$

(iii) $g(X_i)$ has zero mean and finite variance under f,

$$\int_{-\infty}^{\infty} g(z) f(z) dz = 0 \quad (2\text{-}58)$$

and

$$\int_{-\infty}^{\infty} g^2(z) f(z) dz < \infty \quad (2\text{-}59)$$

Assumptions D (i) and (ii) are similar to Assumptions C (i) and (ii) of Section 2.3 for the known signal sequence, and are quite reasonable conditions to impose on the coefficients. Under D (iii) we have that $E\{g(X_i)|H_1\} = 0$, which is not restrictive since any arbitrary g can always be normalized to satisfy this condition, by subtracting from it its mean value under H_1. Finally, the finite variance assumption is also quite reasonable. Notice that the LO nonlinearity g_{LO} for given f does satisfy condition D (iii), for f satisfying Assumptions A and B of Section 2.2.

Now consider the sequence of hypothesis-testing problems $\{H_{1,n}$ versus $K_{1,n}, n = 1, 2, \ldots\}$, with $\theta_n = \gamma/\sqrt{n}$ for some fixed $\gamma > 0$, and the sequence of GC detectors $\{D_{GC,n}, n = 1, 2, \ldots\}$ based on test statistics $T_{GC}(X)$ given by (2-35) and satisfying assumptions D. Under the null hypotheses $H_{1,n}$, it is easy to see that $(\gamma/\sqrt{n}) T_{GC}(X)$ has mean zero and asymptotic variance

$$\lim_{n \to \infty} V\left\{\frac{\gamma}{\sqrt{n}} T_{GC}(X) | H_{1,n}\right\} = \gamma^2 P_a^2 \int_{-\infty}^{\infty} g^2(z) f(z) dz \quad (2\text{-}60)$$

Furthermore, it follows from a central limit theorem that $(\gamma/\sqrt{n}) T_{GC}(X)$ is asymptotically Gaussian under $H_{1,n}$.

To obtain the characteristics under $K_{1,n}$, let us use the expansion

$$\frac{\gamma}{\sqrt{n}} T_{GC}(X) = \frac{\gamma}{\sqrt{n}} \sum_{i=1}^{n} a_i g\left(W_i + \frac{\gamma}{\sqrt{n}} a_i\right)$$

$$= \frac{\gamma}{\sqrt{n}} \sum_{i=1}^{n} a_i g(W_i) + \frac{\gamma^2}{n} \sum_{i=1}^{n} a_i a_i g'(W_i)$$

$$+ o(1) \quad (2\text{-}61)$$

[cf. expansion (2-49)], assuming for the time being the additional regularity conditions required to make this valid. We find from this that

$$\lim_{n \to \infty} E\left\{\frac{\gamma}{\sqrt{n}} T_{GC}(X) | K_{1,n}\right\} = \gamma^2 P_{aa} \int_{-\infty}^{\infty} g'(z) f(z) dz$$

$$= -\gamma^2 P_{aa} \int_{-\infty}^{\infty} g(z) f'(z) dz \quad (2\text{-}62)$$

$$\lim_{n \to \infty} V\left\{\frac{\gamma}{\sqrt{n}} T_{GC}(X) | K_{1,n}\right\} = \gamma^2 P_a^2 \int_{-\infty}^{\infty} g^2(z) f(z) dz \quad (2\text{-}63)$$

and that $(\gamma/\sqrt{n}) T_{GC}(X)$ is also asymptotically Gaussian under $K_{1,n}$. Here the quantity P_{aa} is

$$P_{aa} = \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} a_i a_i \quad (2\text{-}64)$$

This heuristic proof that $(\gamma/\sqrt{n}) T_{GC}(X)$ is asymptotically Gaussian with mean $-\gamma^2 P_{aa} \int g f'$ and variance $\gamma^2 P_a \int g^2 f$ follows a line of reasoning similar to one that we employed for the LO detectors earlier. Note that for $g = g_{LO}$ and $a_i = a_i$, these results reduce to those obtained for the LO detectors. Furthermore, it is also true that the asymptotic normality under $K_{1,n}$, with the mean and variance we have given above can be established without using any regularity assumptions other than Assumptions A, B, C, and D. Such a proof depends on use of LeCam's lemmas, which, as we have indicated earlier, are discussed in Hajek and Sidak [1967, Ch. VI].

Proceeding as we have done before for the LO detectors, we find that for the sequence of GC detectors with asymptotic size α

the asymptotic power function is (Problem 2.3)

$$\lim_{n\to\infty} P_d\left(\frac{\gamma}{\sqrt{n}} \,\Big|\, D_{GC,n}\right)$$
$$= 1 - \Phi\left[\Phi^{-1}(1-\alpha) - \gamma\frac{P_{ss}}{P_s}E(g,f)\right] \tag{2-65}$$

where

$$E(g,f) = \frac{-\int_{-\infty}^{\infty} g(z)f'(z)\,dz}{\left[\int_{-\infty}^{\infty} g^2(z)f(z)\,dz\right]^{1/2}}$$
$$= \frac{\int_{-\infty}^{\infty} g(z)\left[\dfrac{-f'(z)}{f(z)}\right]f(z)\,dz}{\left[\int_{-\infty}^{\infty} g^2(z)f(z)\,dz\right]^{1/2}} \tag{2-66}$$

The function $1 - \Phi[\Phi^{-1}(1-\alpha) - c]$ with value c at $c = 0$ is an increasing function of c with value unity at $c = 0$. Thus for $\gamma > 0$ the limiting detection power will be no less than α if $P_{ss}E(g,f)$ is non-negative. This quite reasonable requirement will always be assumed to hold; the sign of g or of the coefficient sequence can always be picked to meet this requirement. Comparing the limiting power given by (2-52) for the AO detectors and the result (2-65) for the GC detectors, we see that an index of their asymptotic relative performance is provided by the following ratio, which we will call the *asymptotic relative gain* (ARG):

$$ARG_{GC,AO} = \frac{P_{ss}E(g,f)}{P_sP_e\sqrt{I(f)}}$$
$$= \rho_{ss}\frac{E(g,f)}{\sqrt{I(f)}} \tag{2-67}$$

Here ρ_{ss} is the correlation coefficient

$$\rho_{ss} = \frac{P_{ss}}{P_sP_e} \tag{2-68}$$

The coefficient ρ_{ss} attains its maximum value of unity if $a_i = ku_i$, all i, for any $k > 0$. We also find that

$$\frac{E(g,f)}{\sqrt{I(f)}} = \frac{\int_{-\infty}^{\infty} g(z)g_{LO}(z)f(z)\,dz}{\left[\int_{-\infty}^{\infty} g^2(z)f(z)\,dz \int_{-\infty}^{\infty} g_{LO}^2(z)f(z)\,dz\right]^{1/2}} \tag{2-69}$$
$$\leq 1$$

with equality being achieved when $g \equiv g_{LO}$.

Let us view the right-hand sides of (2-52) and (2-65) as approximations for the actual detection probabilities for the AO and GC detectors, respectively, for large sample sizes. Then we may write

$$P_d(\theta \mid D_{GC,n}) \approx 1 - \Phi[\Phi^{-1}(1-\alpha) - \theta\sqrt{n}\,P_s\rho_{ss}E(g,f)] \tag{2-70}$$

and

$$P_d(\theta \mid D_{AO,n}) \approx 1 - \Phi[\Phi^{-1}(1-\alpha) - \theta\sqrt{n}\,P_s\sqrt{I(f)}] \tag{2-71}$$

For fixed n (and P_s, since the signal is known) we see that the $ARG_{GC,AO}$ measures the relative *signal amplitudes* required for the performances of the two detectors to be identical. Specifically, $ARG_{GC,AO}$ is the amplitude of the signal required by the AO detector relative to that required by the GC detector for their detection probabilities to be the same at any common value α of the false-alarm probability, in the asymptotic case of large n.

On the other hand, if we fix θ at some small value, then it follows immediately that the *square* of $ARG_{GC,AO}$ is the $ARE_{GC,AO}$, the ratio of *sample sizes* required by the two detectors for identical asymptotic performance. Thus we have

$$ARE_{GC,AO} = [ARG_{GC,AO}]^2$$
$$= \frac{(P_{ss}^2/P_s^2)E^2(g,f)}{P_e^2 I(f)} \tag{2-72}$$

The numerator in (2-72) may be identified as the *efficacy* ξ_{GC} of the GC detector,

$$\xi_{GC} = \rho_{ss}^2 P_s^2 E^2(g,f) \tag{2-73}$$

which reduces to $P_s^2(f)$ for the AO detector using $s_i = s_i$ and $g = g_{LO}$. Since the signal sequence is known we will henceforth assume that $s_i = s_i$, so that we get $\rho_{ss} = 1$ and

$$ARE_{GC,AO} = \frac{E^2(g,f)}{I(f)}$$
$$= \frac{\xi_{GC}}{I(f)} \quad (2\text{-}74)$$

In general ξ_{GC} is the *normalized efficacy*

$$\xi_{GC} = \frac{\xi_{GC}}{P_s^2}$$
$$= \rho_{ss}^2 \, E^2(g,f) \quad (2\text{-}75)$$

We may extend this result quite easily to obtain the ARE of two GC detectors, D_{GC1} and D_{GC2}, based on nonlinearities g_1 and g_2 and the same coefficients $a_i = s_i$ as

$$ARE_{GC1,GC2} = \frac{\xi_{GC1}}{\xi_{GC2}} \quad (2\text{-}76)$$

where

$$\xi_{GC\,i} = \frac{\left[\displaystyle\int_{-\infty}^{\infty} g_i(x) f'(x)\, dx\right]^2}{\displaystyle\int_{-\infty}^{\infty} g_i^2(x) f(x)\, dx} \quad (2\text{-}77)$$

Notice that the above result for the normalized efficacy of any GC detector could have been derived formally by applying the Pitman-Noether result of (1-41).

In our development above we considered sequences of GC detectors obtained with fixed characteristics g for different sample sizes n. More generally, we can consider sequences of GC detectors with characteristics $g(x,n)$ which are a function of n. One such case is that of (2-55). The quantity $E(g,f)$ is again the asymptotic ratio of the mean to the standard deviation of the GC test statistic under K_1, when $n \to \infty$ (assuming asymptotic normality). Stated loosely, if $g(z,n)$ converges to some fixed $\bar{g}(x)$, then $E(g,f)$ becomes $E(\bar{g},f)$. What we really need is convergence of the mean and variance of $g(X,n)$ as required under the

conditions for Theorem 3 in Section 1.6.

Our results indicate that if our sequence of GC detectors is an AO sequence, than the $ARE_{GC,AO} = 1$. Conversely, if $ARE_{GC,AO} = 1$ and $\xi_{GC} = I(f)$, we find that the sequence of GC detectors is an AO sequence. One conclusion we have drawn from (2-70) and (2-73) is that the efficacy of a detector indicates the absolute power level obtainable from it in the asymptotic case. Modifying (2-70) slightly, we get, with σ^2 the noise variance,

$$P_d(\theta|D_{GC,n}) \approx 1 - \Phi\left[\Phi^{-1}(1-\alpha) - \frac{\theta}{\sigma}\sqrt{n}\ P_s\rho_{ss}\sigma E(g,f)\right] \quad (2\text{-}78)$$

from which we deduce that it is really the quantity $\sigma^2\xi_{GC}$ which gives an indication of the large sample-size power of the detector for any given value of signal-to-noise ratio (SNR) θ/σ. For the LO and AO detectors we have that $\sigma^2\xi_{GC} = P_s^2\sigma^2 I(f)$; thus it is of interest to find the worst-case density minimizing $\sigma^2 I(f)$. It can be shown (Problem 2.4) that the density function minimizing $\sigma^2 I(f)$ is just the Gaussian density function. Thus we may conclude that with optimal processing it is always possible to obtain better performance for non-Gaussian noise than with Gaussian noise under the same SNR conditions. In some cases the potential improvements in performance are quite large.

As an example of the application of these results, consider the LO detector for which $g(x) = x$. From (2-77) we find that the normalized efficacy of the LO detector is $\xi_{LG} = 1/\sigma^2$, where σ^2 is the noise variance. For the sign correlator detector for which $g(x) = sgn(x)$, we have the normalized efficacy

$$\xi_{SO} = \int_{-\infty}^{\infty} sgn(x) f'(x)\, dx \quad (2\text{-}79)$$

$$= 4 f^2(0) \ .$$

Thus we have

$$ARE_{SO,LO} = 4\sigma^2 f^2(0) \quad (2\text{-}80)$$

This has a value of 0.64 for Gaussian noise and a value of 2.0 for noise with the double-exponential pdf.

In concluding this discussion of asymptotic performance, let us observe that the efficacy may be interpreted as an *output SNR* of a detector. Defining the output SNR of a GC detector as

$$SNR_0 = \frac{[E\{T_{GC}(X)|K_1\} - E\{T_{GC}(X)|H_1\}]^2}{V\{T_{GC}(X)|K_1\}} \quad (2\text{-}81)$$

let us consider the case where θ is small. Then it is easily seen that SNR_0 is approximately $n \theta^2 \hat{\rho}_{uu}^2 \hat{P}_u^2 E^2(g,f)$, where $\hat{\rho}_{uu}$ and $\hat{P}_u$ are the finite-n versions of the limiting values ρ_{uu} and P_u. The quantity $\hat{\rho}_{uu}^2 \hat{P}_u^2 E^2(g,f)$ is called the *differential SNR* or *DSNR*, and is essentially the same as the efficacy. A general discussion of second-order measures such as these may be found in [Gardner, 1980].

2.4.2 Finite-Sample-Size Performance

Suppose we wish to compare the exact performances of two GC detectors operating in noise with a given probability density function. Then, as we have noted at the beginning of this section, we need in general to consider all combinations of sample size n, signal-to-noise ratio (or signal amplitude θ), and false-alarm probability p_f, and compute the associated detection probabilities. The detection probabilities can, in principle, always be computed using numerical techniques; in practice this may be costly and tedious. Even after all these computations have been made, no *single* index of relative performance can be given, although specific aspects of the relative performance can be given (e.g., relative amplitudes at given n and p_f, and $p_d = 0.9$) may be so characterized. Of course, the entire procedure has to be repeated for a different noise density function.

While the speed and power of modern computing facilities may make such exhaustive comparisons quite feasible, this type of comparison does not yield simple closed-form expressions for indices of relative performance. The availability of such an index is of prime importance as a basis for initial *design* of a good detection scheme. For example, we will see in Chapter 4 how the efficacy can be used as a basis for optimum quantizer design. The efficacy and the ARE are asymptotic absolute performance and relative performance measures, respectively, which are very convenient to use. What is therefore very important to know is the extent to which the ARE of two detectors is a good indicator of their performance under finite-sample-size and non-zero-signal strength conditions.

The answer to this will obviously depend on the particulars of the two detectors, that is, on their characteristic nonlinearities g; on the noise density function f; and on the combination of sample size n, signal amplitude θ, and false-alarm and detection probabilities p_f, and p_d. Note that in using the efficacy and the ARE we make two key assumptions about our detection problem. One is that the sample size n is large enough so that the test statistic of a GC detector may be assumed to have its asymptotic Gaussian distribution. The other is that the signal amplitude θ is small enough to allow us to use the first-order approximations leading to (2-62). What we would like to have, then, are results

which allow us to make some general conclusions about the validity of these assumptions in finite-sample-size performance comparisons under various conditions.

Within the class of GC detectors the linear correlator detector is widely used as the optimum detector for Gaussian noise. The second most popular detector in this class is the sign correlator (or sign detector), which is the AO detector for noise with the double-exponential noise density function. This simple detector has excellent asymptotic performance characteristics for heavy-tailed noise density functions, of which the double-exponential noise density function is usually taken as a simple representative. Thus it is natural that the relative performances of the LC and SC detectors for Gaussian and double-exponential noise densities are special cases of considerable interest. It turns out that the ARE generally gives a good indication of finite-sample-size relative performance of these detectors for Gaussian noise, but it may considerably over-estimate the advantage of the sign detector in noise with a double-exponential noise density function. We will see that this is primarily caused by the second key assumption in applying the ARE here: that signal amplitude is low enough to allow use of first-order approximations. This is a rather stringent condition and requires very weak signals (and therefore large sample sizes) for noise densities with the "peaked" behavior at the origin when used with the sign detector, whose characteristic nonlinearity g rises by a single step at the origin. In the case of sign detection of a weak signal, behavior of the noise pdf at the origin is of particular significance, so that the weak-signal approximation is more readily met when the noise density function is flatter at the origin, as in the case of Gaussian noise.

The Sign and Linear Detectors

For the sign correlator detector, in the special case of detection of a *constant* signal for which the SC detector coefficients are all unity, it is relatively easy to compute the finite-sample-size performance in additive i.i.d. noise. This is because the test statistic then has a distribution which is always based on a binomial distribution with parameters (n,q). The value of q is known under the null hypothesis (it is $1/2$ for zero-median noise), and with signal present q can be readily determined as $q = P\{W_i > -\theta\}$, which is $1 - F(-\theta)$, where F is the distribution function of the noise.

To compare the finite-sample-size performance of the sign detector with that of another detector we still need to obtain the finite-sample-size performance of the other detector, and this may not be as straightforward. However, in the important special case of comparison of the sign detector with the linear detector for an additive constant signal in Gaussian noise, we are able to perform the comparison with relative ease. The linear detector is simply

double-exponential noise density the *hyperbolic secant* (sech) noise density function defined by

$$f(x) = \frac{1}{2}\operatorname{sech}(\pi x/2) \qquad (2\text{-}82)$$

was considered. This density function has tail behavior similar to that of the double-exponential function, and has behavior near the origin which is similar to that of the Gaussian density function. These properties can readily be established by considering the behavior of the sech function for large and small arguments. Michalsky et al. present their results as a set of plots showing how the efficiency of the sign detector relative to the linear detector behaves for increasing sample sizes. The relative efficiency is defined as a ratio of sample sizes k and $n(k)$ required by the linear and sign detectors, respectively, to achieve the same values for α and p_d. Their procedure was to pick a value for α and set $p_d = 1 - \alpha$, and then to determine the signal amplitude $\theta(k)$ as a function of k required for the linear detector to achieve this performance, for increasing values of k starting from $k = 1$. Then for each k and signal amplitude $\theta(k)$ the number of samples $n(k)$ required by the sign detector for the same α and p_d was computed. For the double-exponential and sech noise pdf's closed-form expressions are available for the density function of the linear detector test statistic; numerical integration was used to perform the computations for the threshold and randomization probabilities and the detection probabilities.

Figure 2.8 is reproduced from the paper by Michalsky et al. It shows the ratio $k/n(k)$, the computed relative efficiency of the sign detector relative to the linear detector, as a function of k for Gaussian noise and $\alpha = 10^{-4}$. Also shown on this figure is a plot of the estimated relative efficiencies based on use of the Gaussian approximation for the sign detector test statistic. We see that for this case where $\alpha = 10^{-4}$ and $p_d = 1 - 10^{-4}$, the relative efficiency is above 0.55, within about 15% of the ARE, for $k \geq 35$. A general conclusion which can be made from the results obtained for this case is that as α decreases, the relative efficiencies converge more slowly to the ARE value. One reason for this may be that the ARE result is based on Gaussian approximations to the distributions of the test statistics (in this case the sign detector statistic). For decreasing α, the threshold setting obtained from the tail region of the Gaussian approximation will be accurate only for larger sample sizes. Another reason is that in this study p_d was set to equal $1 - \alpha$, so that p_d increases as α decreases, requiring larger signal amplitudes. Before we discuss this in more detail let us note some other interesting results obtained in the paper by Michalsky et al.

the linear correlator detector in which the coefficients α_i are again all unity, which is appropriate for the case of additive constant-signal detection. For Gaussian noise the LC detector test statistic has a Gaussian distribution for all sample sizes, making it very easy to compute detection probabilities. This particular finite-sample-size comparison, the sign detector versus the linear detector for for constant signal in Gaussian noise, is of considerable interest. The performance of the widely used linear detector (and more generally, of the LC detector) is the benchmark for this type of detection problem, and performance under Gaussian noise is considered important in evaluating detector performance under the signal-present hypothesis. On the other hand, the sign detector is a *nonparametric* detector of a very simple type, and is able to maintain its design value of false-alarm probability for all noise pdf's with the same value of $F(0)$ as the nominal value (e.g., $F(0) = 1/2$ for zero-median noise). Although one may choose to use detectors such as the sign detector for reasons other than to obtain optimum performance in Gaussian noise, performance under conditions of Gaussian noise, a commonly made if not always valid assumption, is required to be adequate.

Figure 2.7 shows the results of a computation of the power functions for constant-signal detection in additive zero-mean Gaussian noise for the sign and linear detectors. Two different combinations of the sample size ($n = 50, 100$) and false-alarm probability ($\alpha = 10^{-2}, 10^{-3}$) were used to get the two plots. We know that the ARE of the sign detector relative to the linear detector for this situation is 0.64. Do these power function plots allow us to make a quantitative comparison with this asymptotic value of 0.64? We have noted earlier that the ARE can be directly related to the asymptotic SNR *ratio* required to maintain the same detection probability, for the same n and α, for two detectors. The SNR ratio at any given detection probability value is influenced by the *horizontal* displacement of the power curves for the sign and linear detectors. A very approximate analysis based on the power functions for $n = 100$ and $\alpha = 10^{-2}$ shows that at both $p_d = 0.8$ and $p_d = 0.4$, we get an SNR ratio (linear detector to sign detector) of approximately 0.63. The SNR here is defined as θ^2/σ^2. This should be compared with the ARE of 0.64 for this situation. A similar rough check at $p_d = 0.4$ and 0.6 for $n = 50$, $\alpha = 10^{-3}$ reveals a slightly smaller SNR ratio. A very preliminary conclusion one may make from such results is that for the values of n and α we have used, the ARE gives a good quantitative indication of actual finite-sample-size relative performance.

This last conclusion (for Gaussian noise) is borne out by some recent results of a study on the convergence of relative efficiency to the ARE done by Michalsky, Wise, and Poor [1982]. In this study the sign and linear detectors have been compared for three types of noise densities. In addition to the Gaussian and

Figure 2.8 Relative Efficiency of the Sign Detector Compared to the Linear Detector for Gaussian Noise at $\alpha = 10^{-4}$. The Smooth Curve Is Obtained from the Gaussian Approximation. (From [Michalsky et al. 1982])

Figures 2.9(a) and (b) and Figure 2.10 are also reproduced from this paper. Figure 2.9(a) for $\alpha = 10^{-3}$ and double-exponential noise shows a very slow convergence of the relative efficiency of the sign detector relative to the linear detector, to the ARE value of 2.0. It also shows that the relative efficiency can be approximated very well by its estimate based on the Gaussian approximation, with the same means and variances as the exact distributions of the sign and linear detector test statistics. The Gaussian approximation is used in the following way: for each value of k the linear detector test statistic is assumed to be Gaussian with mean $k\theta(k)$ and variance $k\sigma^2$. The threshold is set for $\theta(k) = 0$ using the Gaussian distribution, and the signal strength $\theta(k)$ is then obtained for p_d to equal $1 - \alpha$, again using the Gaussian approximation for the linear detector statistic. This signal strength is now an estimate $\hat{\theta}(k)$ of the correct signal strength. This estimated signal strength is then used to find the binomial parameter $q = P\{W_i > -\theta\}$ needed to find the mean and variance of the sign detector statistic. Finally, again based on the Gaussian approximations for the sign detector test-statistic distributions, the sign detector sample size $n(k)$ is determined to allow it to have the same value for α and $p_d = 1 - \alpha$, with the same signal strength $\hat{\theta}(k)$. Use of the Gaussian approximation allows relative

Figure 2.7 Finite-Sample-Size Performance of Linear and Sign Detectors for Constant Signal θ in Additive Gaussian Noise with Variance σ^2

efficiencies to be easily computed for larger values of k. The result is shown in Figure 2.9(b), where the relative efficiency estimates are shown as a function of both k and $\hat{\theta}(k)$ for values of k beyond the largest shown in Figure 2.9(a). Figure 2.9(b) shows that to obtain a relative efficiency of 1.5 one needs to use more than 600 observations for the linear detector (and more than 400 observations for the sign detector). This rather slow convergence to the ARE value is our first indication that the use of the ARE may not always be justified when one is interested in the finite-sample-size relative performance of two detectors.

The rather different behaviors of the relative efficiencies of the sign and linear detectors for these two noise densities (Gaussian and double-exponential) is not something which could not have been predicted. We have remarked that we make use of two approximations in obtaining the ARE. One is the Gaussian approximation for the distribution of the test statistics. Figure 2.9(a) shows that this is probably quite valid for this particular case. The other approximation we make is that θ is small enough to essentially allow use of the expansion $1 - F(-\theta) \approx 0.5 + \theta f(0)$, where f is the noise density function. For the case of Gaussian noise the density is smooth at the origin and $f'(0) = 0$. In contrast, the double-exponential density is peaked at the origin with non-zero values for the one-sided derivatives there. For the unit-variance double-exponential density we have $f(0) = 1/\sqrt{2}$ and $f'(0^-) = 1$. Thus the signal has to be weaker before the approximation $1 - F(-\theta) \approx 0.5 + \theta f(0)$ becomes valid for the double-exponential density, as compared to the Gaussian density. Consider, for example, the case corresponding to $k = 100$ in Figure 2.9(b) (for $\alpha = 10^{-3}$), for which the signal strength $\hat{\theta}(k)$ is estimated to be 0.62. [This is actually $\hat{\theta}(k)/\sigma$, the variance σ^2 being 1 here.] Let us compare the exact value of $1 - F(-\theta)$ to $0.5 + \theta f(0)$ for $\theta = 0.62$ for the unit variance double-exponential density. A simple computation shows that $1 - F(-\theta) = 0.792$, whereas $0.5 + \theta f(0) = 0.938$, a rather poor approximation. On the other hand, for unit-variance Gaussian noise we get $1 - F(-\theta) = 0.732$ and $0.5 + \theta f(0) = 0.747$ for $\theta = 0.62$, which shows that the approximation is very good for Gaussian noise. For $k = 4000$ we have $\hat{\theta}(k) = 0.1$ from Figure 2.9(b). For $\theta = 0.1$ we find for the unit-variance double-exponential density that $1 - F(-\theta) = 0.560$ and $0.5 + \theta f(0) = 0.571$, which are close. The relative efficiency at $k = 4000$ is seen to be approximately 1.75.

In general we would expect that for any noise density with a local maximum at the origin the relative efficiency of the sign detector relative to the linear detector will be less than its ARE, assuming that k is large enough for the Gaussian approximation to be valid. This is because the approximation $1 - F(-\theta) \approx 0.5 + \theta f(0)$ is then always larger than the correct value of $1 - F(-\theta)$, making the ARE a more optimistic measure of
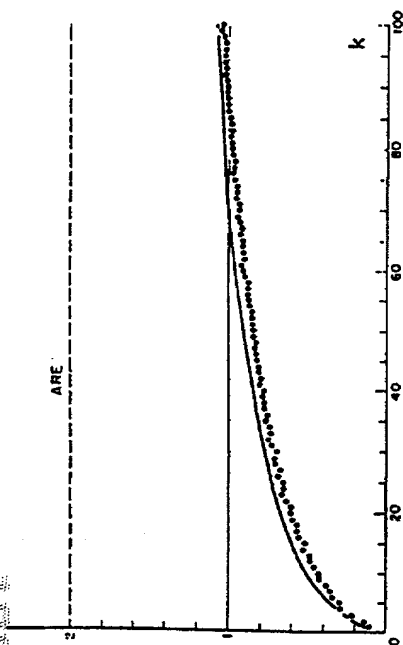
Figure 2.9(a) Relative Efficiency of the Sign Detector Compared to the Linear Detector for Double-Exponential Noise at $\alpha = 10^{-3}$ The Smooth Curve Is Obtained from the Gaussian Approximation. (From [Michalsky et al. 1982])

Figure 2.9(b) The Gaussian Approximation to the Relative Efficiency of the Sign Detector Compared to the Linear Detector for Double-Exponential Noise at $\alpha = 10^{-3}$, as a Function of the Gaussian Approximation $\hat{\theta}(k)$ to the Signal Strength (Normalized by σ) (From [Michalsky et al. 1982])

Figure 2.10 Relative Efficiency Curves for the Sign Detector Compared to the Linear Detector for Sech Noise, for Different Values of α (From [Michalsky et al. 1982])
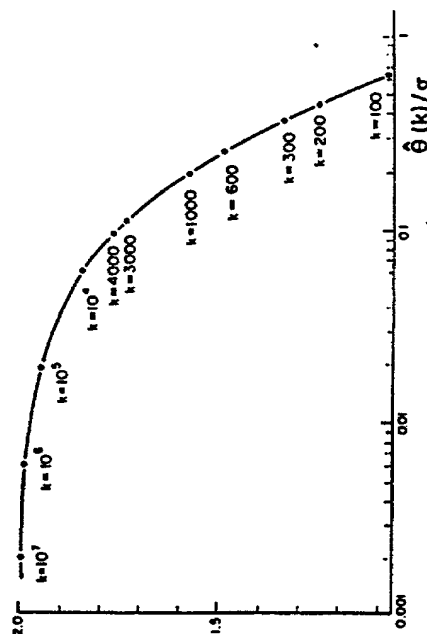
the sign detector relative performance. Since in the study by Michalsky et al. [1982] it was always assumed for computational convenience that $p_4 = 1 - \alpha$, decreasing α leads to increasing p_4 and hence increasing signal strengths for each value of k. Thus it is to be expected that the relative efficiencies will converge more slowly to the ARE as α is decreased.

Figure 2.10 shows a set of plots for different values of α of the relative efficiency of the sign detector relative to the linear detector for the sech noise density function. The ARE value here is 1. The slower convergence rate as α is decreased is apparent in this figure. In addition, we see the expected convergence behavior, which is between that for the double-exponential and Gaussian densities. Prior to the above-described study Miller and Thomas [1975] had also undertaken a numerical study on the convergence of the relative efficiency to the ARE of an asymptotically optimum sequence of detectors compared to the linear detector for the detection of a constant signal in noise with a double-exponential density function. The AO sequence of detectors used was the sequence of GC detectors which are Neyman-Pearson optimum for each n; with increasing n the signal amplitude decreases to maintain a fixed value for the detection probability. The relative efficiency in this numerical study was defined as a ratio of sample sizes required to make the performances of two detectors the same at given values of $p_j = \alpha$ and p_4, for a given value of signal amplitude. The conclusion of Miller and Thomas was also that in this particular situation the relative efficiency can converge rather slowly to its maximum value, which is the ARE value of 2.0. Some further numerical results for this particular case of detection of a constant signal in noise with a double-exponential density function have been given by Marks, Wise, Haldeman, and Whited [1978]. The case of time-varying signal has been considered by Liu and Wise [1983]. More recently Dadi and Marks [1987] have studied further the relative performances of the linear, sign, and optimum detectors for double-exponential noise pdf's and have again reached the same conclusion.

Another type of GC detector which is not too difficult to analyze for finite-sample-size performance is the quantizer-correlator (QC) detector which we shall consider in Chapter 4. In the QC detector the characteristic g of a GC detector is a piecewise-constant quantizer nonlinearity q. For Gaussian noise the performances of QC detectors relative to the LC detector generally follow quite closely the predictions based on asymptotic comparisons. For the double-exponential noise density Michalsky et al. [1982] also considered the simple three-level symmetric QC detector with a middle level of zero, called the *dead-zone limiter* (DZL) detector. Their numerical results indicate that the efficiency of the DZL detector relative to the linear detector for this noise density does converge more rapidly to the ARE value
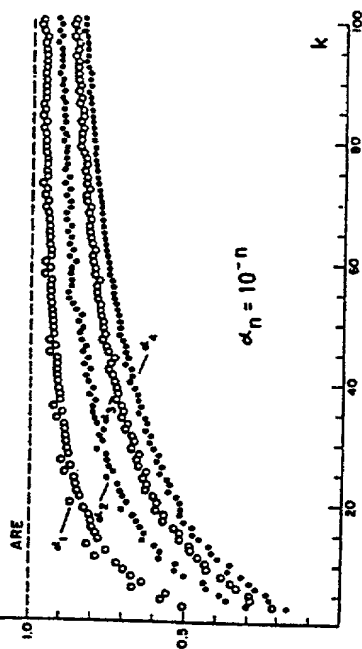
than is the case for the sign detector relative to the linear detector. In fact here one observes very interesting behavior; the relative efficiency actually overshoots the ARE value and then converges to it from above. It is possible to give this overshoot phenomenon an explanation similar to that which explains the convergence from below to the ARE values in our discussion above.

In several other sections in the following chapters we will discuss specific finite-sample-size performance results for particular detectors. Our general conclusion may be summarized as follows. In most cases the use of efficacies and ARE's allows one to make valid qualitative comparisons of different detectors. In many cases the ARE gives a fairly good quantitative indication of finite-sample-size relative performance. In some cases, particularly for the sign detector and pdf's such as the double-exponential noise density function, convergence to the asymptotic value may be quite slow. As a basis for the preliminary design, analysis, and comparison of detection systems, the use of efficacies and ARE's are very convenient and appropriate. As a second step it is highly desirable that some exact finite sample detection performance characteristics be obtained, or that the detector be simulated, to back up conclusions of asymptotic analysis, to fine-tune a design, or to make clear the ranges of applicability of the ARE in performance comparisons. We have also seen that sometimes a simple analysis of the asymptotic approximations made in defining the ARE can give some valuable insights about finite-sample-size relative performance.

2.5 Locally Optimum Bayes Detection

We will now briefly consider the binary signaling problem where one wishes to decide between two known signal sequences in the model given by (2-1), rather than make a choice between noise only and a known signal in noise for the X_i.

Let $\{s_{0i}, i=1,2,...,n\}$ and $\{s_{1i}, i=1,2,...,n\}$ be two known signal sequences of which one is present in the observations, with non-zero amplitude θ. Retaining our assumption of independence and identical distributions for the additive noise components W_i, the likelihood ratio for this hypothesis-testing problem becomes

$$L(X) = \prod_{i=1}^{n} \frac{f(X_i - \theta s_{1i})}{f(X_i - \theta s_{0i})}$$

(2-83)

Let $p_j > 0$ be the a priori probability that the signal sequence $\{s_{ji}, i=1,2,...,n\}$ has been received in noise, $j=0,1$. Of course $p_0 + p_1 = 1$. Let c_{jk} be the cost of deciding that the j-th signal was transmitted when in fact the k-th one was transmitted, for $j,k=0,1$. We will assume that $c_{jj} = 0$ and that $c_{jk} > 0$ for $j \neq k$. Then the Bayes detector minimizing the Bayes risk is one implementing the test

$$L(X) > \frac{c_{10}p_0}{c_{01}p_1}$$

(2-84)

With $L(X)$ exceeding the right-hand side above the signal sequence $\{s_{1i}, i=1,2,...,n\}$ is decided upon.

Denoting by $\exp(k)$ the right-hand side of (2-84), we have the equivalent form for the above test,

$$\sum_{i=1}^{n} \ln f(X_i - \theta s_{1i}) > k + \sum_{i=1}^{n} \ln f(X_i - \theta s_{0i})$$

(2-85)

Now using a Taylor series expansion for $\ln f(z - u)$ about $u = 0$, assuming sufficient regularity conditions on f, we obtain as another equivalent form of the above test,

$$\theta \sum_{i=1}^{n} (s_{1i} - s_{0i}) g_{LO}(X_i)$$
$$+ \frac{\theta^2}{2} \sum_{i=1}^{n} (s_{1i}^2 - s_{0i}^2) \left[\frac{f''(X_i)}{f(X_i)} - g_{LO}^2(X_i) \right] + o(\theta^2) > k$$

(2-86)

where g_{LO} has been defined in (2-33).

This result allows some interesting interpretations. For a finite sample size n, in the case of vanishing signal strengths (the local case), by ignoring second-order and higher-order terms we find that the Bayes detector implements the locally optimum Bayes test

$$\theta \sum_{i=1}^{n} (s_{1i} - s_{0i}) g_{LO}(X_i) > k$$

(2-87)

Notice that the test statistic on the left-hand side above contains the signal amplitude θ explicitly. Suppose, for example, that k is positive because $c_{10} > c_{01}$ and $p_0 = p_1$ in (2-84). This means that the error of falsely deciding that the $s_{1i}, i=1,2,...,n$, were transmitted is more costly than the other type of error. Then for a given set of observations $X_1, X_2, ..., X_n$, the assumed small value of $\theta << \sigma$ will influence the final decision made by the detector. As θ becomes smaller the observations become less reliable as an objective means for discriminating between the two signal alternatives, and the decision which is less costly when no observations are available tends to be made by the detector.

In the special but commonly assumed case of equal costs and prior probabilities we have $k = 0$ in (2-87) and θ can then be dropped from the left-hand side. The resulting locally optimum Bayes detector than has the structure of the LO detector based on the Neyman-Pearson criterion, with the threshold now fixed to be zero.

In considering the asymptotic situation $n \to \infty$ and $\theta \to 0$ simultaneously, we have to specify the relationship between θ and n. Let us assume, as we have done in Section 2.3, that $\theta = \gamma/\sqrt{n}$ for some fixed $\gamma > 0$. Further, we will assume that the signal sequences have finite average powers,

$$0 < \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} s_{ji}^2 = P_j^2 < \infty$$

(2-88)

for $j = 0,1$. Under these conditions we find that the second plus higher-order term on the left-hand side of (2-86) converges in probability under both alternatives to a constant which is

$$\lim_{n \to \infty} \frac{\gamma^2}{2n} \sum_{i=1}^{n} (s_{1i}^2 - s_{0i}^2) \left[\frac{f''(X_i)}{f(X_i)} - g_{LO}^2(X_i) \right]$$
$$= -\frac{\gamma^2}{2} (P_1^2 - P_0^2) I(f)$$

(2-89)

Here $I(f)$ is the Fisher information for location shift for the noise pdf f.

Using this result we obtain the locally optimum Bayes test, which retains its optimality under the specific asymptotic situation $n \to \infty$, $\theta \to 0$, and $\theta\sqrt{n} \to \gamma > 0$ to be

$$\theta \sum_{i=1}^{n} (s_{1i} - s_{0i}) g_{LO}(X_i) > k + \frac{\gamma}{2} (P_1^2 - P_0^2) I(f) \quad (2\text{-}90)$$

The effect of letting θ approach zero for a fixed, finite n is to make $\gamma \to 0$ (set $\gamma = \theta\sqrt{n}$), which gives us the locally optimum Bayes detector. We find that when $P_1^2 = P_0^2$ the locally optimum Bayes detector employing the test of (2-87) does retain its optimality asymptotically. Otherwise, the second "bias" term has to be added to the threshold in (2-90) as a necessary condition for asymptotic optimality.

The important practical result from the above development is that for the commonly assumed case of equal costs for the two types of errors, equal *a priori* probabilities for the two alternatives about the signal, and antipodal signaling ($s_{1i} = -s_{0i}$, $i = 1,2,...,n$), the locally optimum Bayes detector is based on the test

$$\sum_{i=1}^{n} (s_{1i} - s_{0i}) g_{LO}(X_i) > 0 \quad (2\text{-}91)$$

This also yields an asymptotically optimum sequence of Bayes detectors. More generally a non-zero threshold, modified by an additive bias term, and normalized by dividing with the signal amplitude θ, is required.

Our results for the locally and asymptotically optimum Bayes detectors required particular attention to be focused on the "bias" term in the threshold and on the signal amplitude θ which is in general necessary to obtain the correct threshold normalization. This is to be contrasted with the LO and AO detectors based on the Neyman-Pearson criterion. There the threshold setting is decided at the end to obtain the required false-alarm probability, any simplifying monotone transformations of the test statistic being permissible in the intermediate steps in deriving the final structure of the threshold test.

The derivation and analysis of locally and asymptotically optimum Bayes detection schemes for known binary signals has been considered by Middleton [1966] and also more recently by him [1984]. This latter paper, as well as Spaulding and Middleton [1977a], give detection performance results based on upper bounds

on the error probabilities and also based on the asymptotic Gaussian distributions of the test statistic, for noise pdf's modeling impulsive noise. Some performance results have also been given by Spaulding [1985] for the LO Bayes detector. Maras, Davidson, and Holt [1985a] give detection performance results for the Middleton Class A noise model which we shall discuss briefly in the next chapter. An interesting sub-optimum GO detection scheme has also been suggested by Hug [1980].

2.6 Locally Optimum Multivariate Detectors

In this last section we will consider a generalization of our basic model (2-1) for the univariate observations X_i that we have been concerned with so far. We will here study the case where we have a set of i.i.d. *multivariate* or *vector* observations $Y_i = (X_{1i}, X_{2i}, ..., X_{Li})$ described by

$$Y_i = \theta r_i + V_i, \quad i = 1,2,...,n \quad (2\text{-}92)$$

Now the $r_i = (s_{1i}, s_{2i}, ..., s_{Li})$ are known L-variate signal vectors and the $V_i = (W_{1i}, W_{2i}, ..., W_{Li})$ are, for $i = 1,2,...,n$, a set of i.i.d. random noise vectors with a common L-variate density function f.

Such a model for multivariate observations can describe the outputs of L receivers or sensors forming an array, designed to pick up a signal from a distant source in a background of additive noise. At each sampling time the L sensor outputs are obtained simultaneously. The noise in each sensor may be correlated with noise in other sensors at any one sampling instant, although our model above requires the noise samples to be temporally independent. The signal arrives as a propagating plane wave from a particular direction, so that the components of each r_i are delayed versions of some common signal waveform in this type of problem.

Another application in which the model is appropriate is that in which a scalar observed waveform possibly containing a known signal is sampled in periodic bursts or groups of L closely spaced samples. If sufficient time is allowed between each group of L samples the groups may be assumed to be independent, although the samples within each group may be dependent. Such a scheme can be used to improve the performance of a detector which is constrained to operate on independent samples only, allowing it to use more data in the form of independent multivariate samples. We shall see in Chapter 5 that for detection of a narrowband signal which is completely known in additive narrowband noise we can interpret the in-phase and quadrature observation components as having arisen from such a sampling scheme with $L = 2$.

Under the signal-present hypothesis the density function f_Y of the set of n observation vectors $Y = (Y_1, Y_2,..., Y_n)$ is now

$$f_{Y|Y}(y|\theta) = \prod_{i=1}^{n} f(y_i - \theta r_i)$$ (2-93)

keeping in mind that f is now a function of an L-component vector quantity. The LO detector now has a test statistic

$$\lambda_{LO}(Y) = \frac{\frac{d}{d\theta} f_Y(Y|\theta)\big|_{\theta=0}}{f_Y(Y|0)}$$

$$= \frac{\sum_{i=1}^{n} \frac{d}{d\theta} f(Y_i - \theta r_i)\big|_{\theta=0}}{f(Y_i)}$$

$$= \sum_{i=1}^{n} r_i \left[\frac{-\nabla f(Y_i)}{f(Y_i)}\right]^T$$ (2-94)

where ∇f is the gradient vector defined by

$$\nabla f(v) = \left[\frac{\partial f(v)}{\partial v_1}, \frac{\partial f(v)}{\partial v_2}, \ldots, \frac{\partial f(v)}{\partial v_L}\right]^T$$ (2-95)

This result reduces to our result of (2-32) for $L = 1$. In our more general case here the locally optimum vector nonlinearity is

$$g_{LO}(y_i) = \frac{-\nabla f(y_i)}{f(y_i)}$$

$$= -\nabla \ln f(y_i)$$ (2-96)

We see that if $f(v) = \prod_{i=1}^{L} f(v_j)$ the test statistic of (2-94) is simply our earlier LO statistic of (2-32), applied to nL independent observations.

Consider a multivariate GC test statistic

$$T_{GC}(Y) = \sum_{i=1}^{n} a_i g^T(Y_i)$$ (2-97)

where $g(y_i) = [g_1(y_i), g_2(y_i),..., g_L(y_i)]$, each g_j being a real-valued function of an L-vector. It can be shown quite easily that the efficacy of such a test statistic for our detection problem is

$$\xi_{GC} = \lim_{n \to \infty} \frac{\left[\sum_{i=1}^{n} a_i \int_{R^L} g^T(v) g_{LO}(v) f(v) dv \, r_i^T\right]^2}{n \sum_{i=1}^{n} a_i \int_{R^L} g^T(v) g(v) f(v) dv \, a_i^T}$$ (2-98)

With $g = g_{LO}$ and $a_i = r_i$ we get the maximum efficacy

$$\xi_{GC,LO} = \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} r_i \, I(f) \, r_i^T$$ (2-99)

where $I(f)$ is the matrix defined by

$$I(f) = E\{g_{LO}(V_i) g_{LO}(V_i)\}$$ (2-100)

which is Fisher's information matrix corresponding to the function $I(f)$ of (2-12) in the univariate case.

Suppose that f is the multivariate Gaussian density,

$$f(v) = \frac{1}{(2\pi)^{L/2}(\det\Lambda)^{1/2}} e^{-v\Lambda^{-1}v^T/2}$$ (2-101)

where Λ is the covariance matrix. Then we find that

$$g_{LO}(y_i) = y_i \Lambda^{-1}$$ (2-102)

and the LO test statistic is the matched filter statistic

$$\lambda_{LO}(Y) = \sum_{i=1}^{n} r_i \Lambda^{-1} Y_i^T$$ (2-103)

In Chapter 5 we will be considering essentially LO test statistics for the bivariate case with the bivariate density function f being *circularly symmetric* so that $f(v_1, v_2) = h(\sqrt{v_1^2 + v_2^2})$. A special case of this is the bivariate Gaussian density for two i.i.d. Gaussian components. The model for multivariate observations that we have treated here has been used by Martinez, Swaszek, and Thomas [1984] to obtain locally optimum detectors.

PROBLEMS

Problem 2.1

Verify that the maximum likelihood estimate $\hat{\theta}_{ML}$ of θ under K_1 of (2-14) for the double-exponential noise pdf is obtained as a solution of (2-24).

Problem 2.2

Draw a block diagram for an implementation of the test statistic $\lambda(X)$ of (2-22) and (2-23), using multipliers, adders, accumulators, and a fixed nonlinearity.

Problem 2.3

Show that the limiting value of the power of a sequence of size α GC detectors for $H_{1,n}$ versus $K_{1,n}$ of (2-42) with $\theta_n = \gamma/\sqrt{n}$ is given by (2-65).

Problem 2.4

Prove that the Gaussian pdf minimizes $\sigma^2 I(f)$, where $I(f)$ is the Fisher information for location, from among all absolutely continuous pdf's.

Problem 2.5

Consider the GC detector based on the soft-limiter nonlinearity

$$g(x) = \begin{cases} c & , & x > c \\ x & , & |x| \le c \\ -c & , & x < -c \end{cases}$$

for some constant $c \ge 0$. Obtain the ARE of this GC detector relative to the LC detector (both using the same GC coefficients) for H_1 versus K_1 of (2-13) and (2-14), when the pdf f is symmetric about the origin. Obtain the limiting values of the ARE for $c \to 0$ and $c \to \infty$. Compute the ARE for $c = 3\sigma$ with f the Gaussian density with variance σ^2. Comment on the implication of this result.

Problem 2.6

Consider the GC detector based on the dead-zone limiter nonlinearity

$$g(x) = \begin{cases} 1 & , & x > c \\ 0 & , & |x| \le c \\ -1 & , & x < -c \end{cases}$$

for some constant $c \ge 0$. Obtain the ARE of this GC detector relative to the LC detector for symmetric f in H_1 versus K_1 of (2-13) and (2-14). Obtain the limiting value for the ARE when $c \to 0$. Find the optimum value of c maximizing the ARE for Gaussian f, and determine the maximum ARE value.

Problem 2.7

The "hole-puncher" function

$$g(x) = \begin{cases} x & , & |x| \le c \\ 0 & , & |x| > c \end{cases}$$

may be used as the detector nonlinearity for a GC detector. Obtain the efficacy of the resulting detector for f the Cauchy pdf

$$f(x) = \frac{1}{\pi\sigma} \frac{1}{1 + (x/\sigma)^2}$$

in testing H_1 versus K_1 of (2-13) and (2-14), and find the optimum value of c/σ maximizing this efficacy. What is the limiting value of the efficacy as $c \to \infty$? Explain your result.

Problem 2.8

The characteristic g of a GC detector is a linear combination of functions in a set $\{g_1, g_2, ..., g_M\}$ which is orthonormal with respect to the weighting function f, the noise pdf. Show that the efficacy of an AO GC detector of this type is the sum of the efficacies of detectors based on the individual g_i, $i = 1, 2, ..., M$.

Problem 2.9

Develop a possible explanation for the overshoot phenomenon mentioned at the end of Section 2.4 in our discussion of numerical results on convergence of relative efficiencies to the ARE for the dead-zone limiter detector relative to the linear detector.

Chapter 3

SOME UNIVARIATE NOISE PROBABILITY DENSITY FUNCTION MODELS

3.1 Introduction

In this chapter we will apply the results we have derived so far to some particular models for univariate non-Gaussian noise probability density functions. The models we will consider have been found to be appropriate for modeling non-Gaussian noise in different systems. A large number of investigations have been carried out over the course of the last forty years on the characteristics of noise processes encountered in different environments. In particular, there has been much interest in characterization of underwater acoustic noise, urban and man-made RF noise, low-frequency atmospheric noise, and radar clutter noise. For such situations it is well-recognized that the mathematically appealing simple Gaussian noise model is often not at all appropriate. It is not within the scope of this book to detail the development of the non-Gaussian noise models that we will describe here, and we will have to refer the reader to the relevant literature to gain an understanding of the basis for the models.

There is one common feature, however, that all these models share. This is that they specify noise density functions which in their tails decay at rates lower than the rate of decay of the Gaussian density tails. The essential implication of this fact is that non-Gaussian noise which tends to produce large-magnitude observations more likely to produce large-magnitude observations than would be predicted by a Gaussian model (satisfying the same constraint on some scale characteristic, such as a noise percentile). We should then expect the optimum and locally optimum detectors to employ some nonlinear characteristic g tending to reduce the influence of large-magnitude observations on the test statistic. This should be expected because large-magnitude observations will now be less reliable (compared to the Gaussian case) in indicating the absence or presence of a signal. We will, indeed, observe this general characteristic.

As an example illustrating the above statements, let us briefly examine some results which have been reported for the noise affecting ELF (extremely low frequency) communication systems. In studies described by Evans and Griffiths [1974], the non-Gaussian atmospheric noise caused by lightning activity in local and distant storms is identified as the major limiting factor in long-range ELF communication systems operating in the range 3 to 300 Hz. The characteristic feature of noise in this band is the

occurrence of impulse-like components, in a background noise waveform due to low attenuation of numerous distant atmospheric noise sources. A typical recording of a noise waveform in this band of frequencies is shown in Figure 3.1. Figure 3.2 shows a typical amplitude probability distribution function for such noise, obtained from empirical data. For comparison, Figure 3.2 also shows a Gaussian distribution function with the same variance. Evans and Griffiths [1974] also present an interesting graph showing the locally optimum nonlinearity g_{LO} obtained for an empirically determined noise probability density function; this is given here as Figure 3.3. Further details on receiver structures employing such nonlinear processing for ELF systems have been given by Bernstein, McNeil, and Richer [1974] and Rowe [1974]. An overview of ELF communication systems may be found in [Bernstein et al., 1974]. Modestino and Sankur [1974] have given some specific models and results for ELF noise. More recently, several suboptimal detectors as well as the locally optimum detector have been studied by Ingram [1984], who obtained detection performance characteristics for ELF atmospheric noise.

We will obtain nonlinear characteristics of the nature of that shown in Figure 3.3 for the types of noise models we will now consider analytically. The first type of model we will consider defines noise density functions as generalizations of well-known univariate densities, one of them the Gaussian density. These have been found to provide good fits in some empirical studies. The second case we will consider is that of mixture noise, which can be given a statistical as well as a physical basis. It will also be shown that a useful noise model developed by Middleton is closely related to the mixture model. We conclude this chapter with a brief discussion of strategies for adaptive detection when the noise probability density function cannot be assumed to be *a priori* known.

3.2 Generalized Gaussian and Generalized Cauchy Noise

The convenient mathematical properties of the Gaussian probability density function have to be given up, at least partially, in order to obtain density function models which are better descriptions of the noise and interference encountered in many real-world systems. As we have remarked above, the presence of impulsive components tends to produce noise density functions with heavier tails than those of the Gaussian densities. One way to obtain such modified tail behavior is to start with the Gaussian density function and to allow its rate of exponential decay to become a free parameter. In another approach we start at the other extreme case of algebraic tail behavior of the Cauchy noise probability density function, and introduce free parameters to allow a range of possibilities which includes the Gaussian as a special case.

3.2.1 Generalized Gaussian Noise

A *generalized Gaussian* noise density function is a symmetric, unimodal density function parameterized by two constants, the variance σ^2 and a rate-of-exponential-decay parameter $k > 0$. It is defined by

$$f_k(z) = \frac{k}{2A(k)\Gamma(1/k)} e^{-|z/A(k)|^k} \qquad (3\text{-}1)$$

where

$$A(k) = \left[\sigma^2 \frac{\Gamma(1/k)}{\Gamma(3/k)}\right]^{1/2} \qquad (3\text{-}2)$$

and Γ is the gamma function:

$$\Gamma(a) = \int_0^\infty z^{a-1} e^{-z}\, dz \qquad (3\text{-}3)$$

We find that for $k=2$ we get the Gaussian density function, and for lower values of k the tails of f_k decay at a lower rate than for the Gaussian case. The value $k=1$ gives us the double-exponential noise density function. Figure 3.4 shows the shapes of $f_k(z)$ for different k, with the variance $\sigma^2 = 1$ for all k. Thus the class of generalized Gaussian density functions allows us to conveniently consider a spectrum of densities ranging from the Gaussian to those with relatively much faster or much slower rates of exponential decay of their tails. Algazi and Lerner [1964] indicate that generalized Gaussian densities with k around 0.5 can be used to model certain impulsive atmospheric noise.

For the generalized Gaussian noise densities we find that the locally optimum nonlinearities g_{LO} are, from (2-33),

$$g_{LO}(z) = \frac{k}{[A(k)]^k} |z|^{k-1}\text{sgn}(z) \qquad (3\text{-}4)$$

These are, as expected, odd-symmetric characteristics, since the f_k are even functions. Note that for $k=2$ we get $g_{LO}(z) = z/\sigma^2$, the result of (2-36), and for $k=1$ we get $g_{LO}(z) = (\sqrt{2}/\sigma)\text{sgn}(z)$, which agrees with the result of (2-38). Figure 3.5 shows the normalized versions $([A(k)]^k/k)g_{LO}(z)$ for the generalized Gaussian densities. Observe that the value $k=1$ separates two distinct types of behavior. For $k>1$ the nonlinearities g_{LO} are continuous at the origin and increase monotonically to $+\infty$ for increasing z, whereas for $k<1$ there is an infinite discontinuity at the origin and g_{LO} decreases monotonically to zero as z increases from zero.

It is quite easy to establish that the Fisher information $I(f_k)$ for the generalized Gaussian density function is

$$I(f_k) = \frac{k^2\Gamma(3/k)\,\Gamma(2-1/k)}{\sigma^2\Gamma^2(1/k)} \qquad (3\text{-}5)$$

which is finite for $k > 0.5$. We can now compare the performance of the LC detector, which is optimum for Gaussian noise, with that of the LO detector based on g_{LO} of (3-4) when the noise density is the generalized Gaussian density f_k. From (2-77) we easily obtain the normalized efficacy $\bar{\xi}_{LC}$ of the LC detector for any noise density function f to be

$$\bar{\xi}_{LC} = \frac{1}{\sigma^2} \qquad (3\text{-}6)$$

where σ^2 is the noise variance. Then we find that the ARE of the LO detector for noise pdf f_k, relative to the LC detector, computed for the situation where f_k is the noise density function, is [from (2-76)]

$$ARE_{LO,LC} = 1/ARE_{SC,LO}$$
$$= \sigma^2 I(f_k)$$
$$= \frac{k^2\Gamma(3/k)\,\Gamma(2-1/k)}{\Gamma^2(1/k)} \qquad (3\text{-}7)$$

For $k=2$ we have, of course, $ARE_{LO,LO}=1$, and $k=1$ gives us $ARE_{SC,LC}=2$ for double-exponential noise. Figure 3.6 shows $ARE_{LO,LO}$ as a function of k. We emphasize that this is the ARE of the LO detector for f_k relative to the LO detector, when f_k is the noise density. Thus explicit knowledge of the non-Gaussian nature of the noise density function can be exploited to get significantly more efficient schemes than the LC detector.

As an example of the comparison of two *fixed* nonlinearities, let us consider $ARE_{SC,LC}$ as a function of k for the generalized Gaussian noise pdf given by (3-1). Applying the result (2-76), we conclude easily that

$$ARE_{SC,LO} = 4\sigma^2 f_k^2(0)$$
$$= \frac{k^2\Gamma(3/k)}{\Gamma^2(1/k)} \qquad (3\text{-}8)$$

Because the sign correlator (SC) detector is the LO detector for $k=1$, we find that $ARE_{SC,LC}$ agrees with $ARE_{LO,LC}$ of (3-7) for
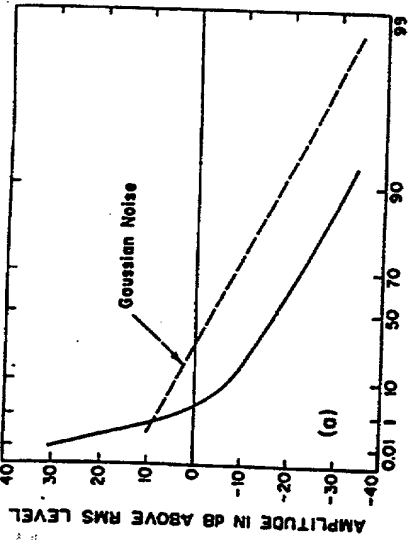
Figure 3.2 Amplitude Probability Distribution of Typical ELF Noise (from Evans and Griffiths [1974]) © 1974 IEEE
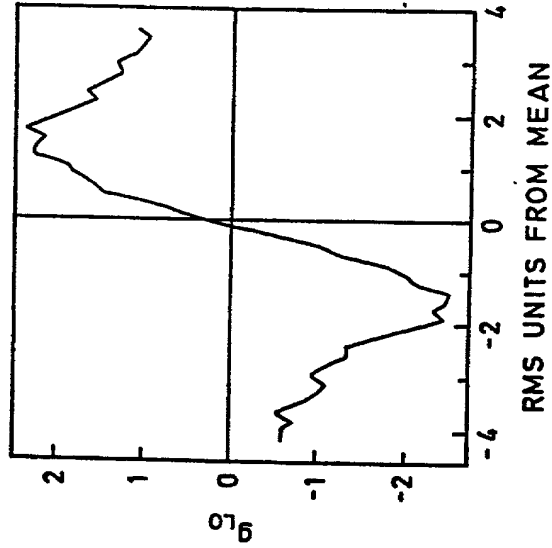
Figure 3.3 Locally Optimum Nonlinearity for Empirically Determined ELF Noise PDF (from Evans and Griffiths [1974]) © 1974 IEEE

$k = 1$. Otherwise, $ARE_{SC,LC}$ is less than $ARE_{LO,LC}$. Nonetheless, as shown in Figure 3.6, the simple SC detector is significantly more efficient that the LC detector for all $k \leq 1$.
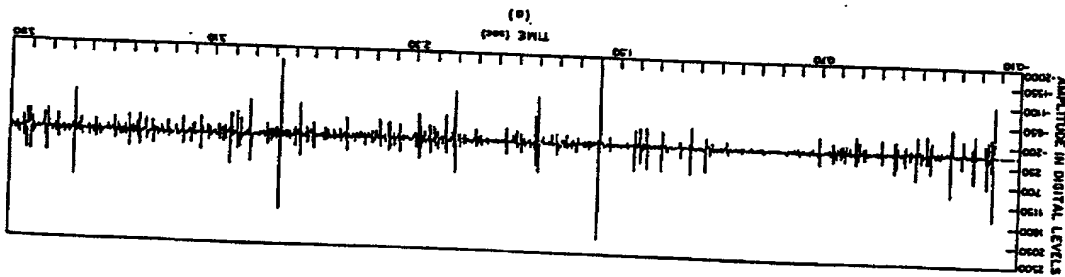
Figure 3.1 Sample Function of Typical ELF Noise Process (from Evans and Griffiths [1974]) © 1974 IEEE

Figure 3.5 Normalized LO Nonlinearities for Generalized Gaussian PDF's

the density function becomes the Cauchy density,

$$f_{k,\nu}(z) = \frac{1}{\pi\sigma}\frac{1}{1+(z/\sigma)^2} \qquad (3\text{-}11)$$

showing that σ^2 is in general a scale parameter but *not* the variance.

The Cauchy density itself does not have a finite variance. In spite of this non-physical property, it has been considered to be useful in modeling impulsive noise. One justification for this is that if a detector has acceptable performance in such noise, then it will most likely have acceptable performance in actual impulsive noise. In addition to being a generalization of the Cauchy noise density, the generalized Cauchy density of (3-9) includes as a special case a model for impulsive noise proposed by Mertz [1961]. From the amplitude density function of the impulsive noise proposed by Mertz, an assumption of symmetry of the noise density function leads to the generalized Cauchy density with $k = 1$. Furthermore, a model for impulsive noise proposed by Hall [1966]

Figure 3.4 Generalized Gaussian Probability Density Functions ($\sigma^2 = 1$)

3.2.2 Generalized Cauchy Noise

The other class of noise density functions which is useful in studying the shapes of the nonlinearities g_{LO} for a range of noise characteristics is the class of *generalized Cauchy* densities. A generalized Cauchy density is defined in terms of three parameters σ^2, $k > 0$ and $\nu > 0$ by

$$f_{k,\nu}(z) = \frac{B(k,\nu)}{\left\{1+\frac{1}{\nu}\left[\frac{|z|}{A(k)}\right]^k\right\}^{\nu+1/k}} \qquad (3\text{-}9)$$

where

$$B(k,\nu) = \frac{k\nu^{-1/k}\Gamma(\nu+1/k)}{2A(k)\Gamma(\nu)\Gamma(1/k)} \qquad (3\text{-}10)$$

and where $A(k)$ is defined by (3-2). The density function $f_{k,\nu}(z)$ has an algebraic rather than an exponential tail behavior. We see from (3-9) that the tails of the density function decay in inverse proportion to $|z|^{\nu k+1}$ for large $|z|$. For $k = 2$ and $\nu = 1/2$

Figure 3.7 Generalized Cauchy Probability Density Functions ($\sigma^2 = 1$, $v = 4$)

The locally optimum nonlinearity g_{LO} for generalized Cauchy noise is easily found to be

$$g_{LO}(z) = \frac{vk+1}{v[A(k)]^k + |z|^k}\, |z|^{k-1}\operatorname{sgn}(z) \qquad (3\text{-}12)$$

which should be compared with g_{LO} for the generalized Gaussian densities of (3-4). The nonlinearity of (3-12) does approach, pointwise, that of (3-4) as $v \to \infty$. On the other hand, for any fixed set of values of the parameters v, k, and σ^2 the characteristic g_{LO} of (3-12) always approaches zero for $|z| \to \infty$. This behavior is illustrated in Figure 3.8, which shows the normalized versions $\{v[A(k)]^k/(vk+1)\}g_{LO}$ of g_{LO} for the generalized Cauchy densities for different values of the parameter k with $\sigma^2 = 1$ and $v = 10$. As we have remarked, for $v = 1/2$ and $k = 2$ we get the Cauchy noise density of (3-11), for which g_{LO} is

$$g_{LO}(z) = \frac{2z}{\sigma^2 + z^2} \qquad (3\text{-}13)$$

Figure 3.6 Asymptotic Relative Efficiencies of the LO Detectors and of the SC Detector, Relative to the LC Detector, for Generalized Gaussian Noise

is obtained with $k = 2$. When $2v$ is an integer this gives a scaled Student-t density.

It can be shown that the variance of the generalized Cauchy density, if it exists, is $\sigma^2 v^{2/k}\Gamma(v-2/k)/\Gamma(v)$. This is finite for $vk > 2$, which is consistent with our observation that the tails of the density function decay in inverse proportion to $|z|^{vk+1}$. Figure 3.7 shows the shapes of the generalized Cauchy densities for different values of k when $v = 4$ and $\sigma^2 = 1$. For a fixed value of σ^2 and k, the limiting case $v \to \infty$ gives the generalized Gaussian density of (3-1). This can be inferred directly from the definition $e^s = \lim_{t \to \infty} (1 + s/t)^t$. Thus we also get the Gaussian density as a limiting case for $k = 2$ and $v \to \infty$. The generalized Cauchy density function is therefore capable of modeling a wide range of noise density types.

This function is plotted in Figure 3.9 for $\sigma^2 = 1$. One interesting conclusion we can draw from these plots is that there are several combinations of the parameters k and v which lead to LO nonlinearities of the type shown in Figure 3.3 for ELF noise; they are approximately linearly increasing near $z = 0$, and then decay to zero for large magnitudes of z.
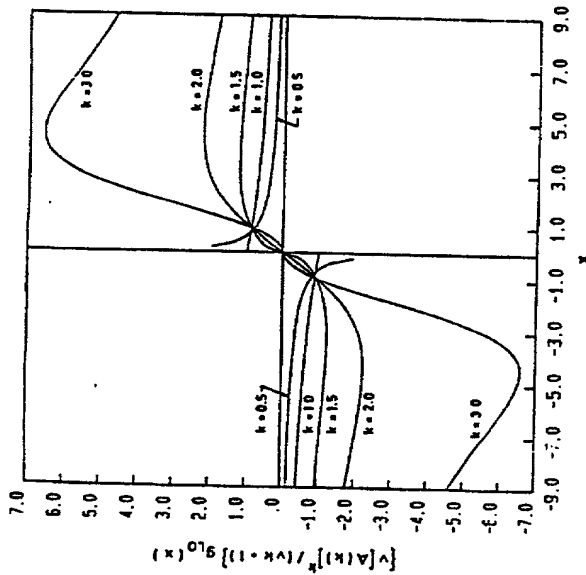
Figure 3.8 Normalized LO Nonlinearities for Generalized Cauchy PDF's ($\sigma^2 = 1$, $v = 10$)

The Fisher information $I(f_{k,v})$ for generalized Cauchy noise can be shown to be

$$I(f_{k,v}) = \frac{(vk+1)^2\,\Gamma(3/k)\,\Gamma(v+1/k)\,\Gamma(v+2/k)\,\Gamma(2-1/k)}{\sigma^2 v^{2/k}\,\Gamma(1/k)\,\Gamma(v)\,\Gamma(2+v+1/k)} \quad (3\text{-}14)$$

a finite quantity for $k > 0.5$. Note that σ^2 is not the variance of $f_{k,v}$ of (3-9); the variance is finite for $vk > 2$. From this we may calculate $ARE_{LO,LC}$, the ARE of the LO detector for $f_{k,v}$ relative to the LC detector, when $f_{k,v}$ is the noise density function. The result is shown in Figure 3.10, where $ARE_{LO,LC}$ is plotted as a function of k for different values of the parameter v. Once again
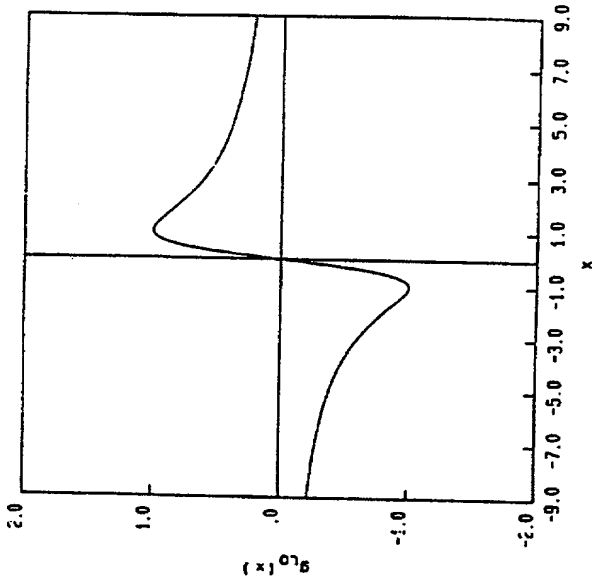
Figure 3.9 LO Nonlinearity for Cauchy Noise PDF ($\sigma^2 = 1$)

we find that substantial improvements in asymptotic performance can be obtained by use of LO or AO detectors if the noise density function is known to be a particular non-Gaussian density. The use of the simple hard-limiting SC detector can also be shown to produce a significant improvement over the LC detector for a range of values of the parameters k and v.

Before we go on to consider mixture noise densities, we should mention that one simple model which leads to some of the noise densities we have discussed above is the random-power-level Gaussian model. Here we model the noise samples W_i as products of standard Gaussian (zero-mean, unit-variance) components and random amplitude factors with various density functions. This is, in fact, the basis for the Hall model. Such noise models have also been used by Spooner [1968], Hatsell and Nolte [1971] and Adams and Nolte [1975]. The results we have described in this section were originally developed by Miller and Thomas [1972], who also considered the finite-support generalized beta densities which we have not included here. Some simple asymmetrical noise pdf models related to the Gaussian pdf are given in [Kanefsy and Thomas, 1965]. Huang and Thomas [1983] have also specifically

behavior of f near the origin is dominated by that of η, assuming that h is a bounded function. For large values of $|z|$, however, h dominates the behavior of f since its tails decay at a slower rate than do those of η.

Mixture densities of the form of (3-15) have been used by many investigators to model heavy-tailed non-Gaussian noise density functions. In robustness studies they have been used to model classes of allowable noise density functions which are in the neighborhood of a nominal Gaussian density function η. The mixture model has also been found to provide a good fit, in many cases, to empirical noise data. As an example, we point to the work by Trunk [1976] in which the mixture model is used to fit the empirically determined probability distribution function of the envelope of radar clutter returns. In this case η in (3-15) is a Rayleigh rather than a Gaussian density. In particular, Trunk used what we call the *Gaussian-mixture* model (more accurately, Rayleigh mixture), with h in (3-15) also a Gaussian (or Rayleigh) density with a variance larger than that of η.

The mixture noise density model of (3-15) can also be given a justification as an appropriate model for impulse noise, considered to be a train of random-amplitude and randomly occurring narrow pulses in a background of Gaussian noise. Let us express such an impulsive component of a noise waveform as

$$I(t) = \sum_{t=-\infty}^{\infty} A_t\, p(t - t_t) \tag{3-16}$$

Here the amplitudes A_t can be taken to be i.i.d. amplitudes and the t_t have commonly been assumed to be generated by a Poisson point process. The pulse shape p is determined by the receiver filter response. The classical analysis of Rice [1944] leads to a general result for the first-order probability density function of $I(t)$. Let υ be the rate parameter of the Poisson point process and let T_p be the width of the pulse p. Then for $\upsilon T_p \ll 1$, one can derive the following approximation for the density function f_I of samples of $I(t)$ [Richter and Smits, 1971]:

$$f_I(z) = (1 - \upsilon T_p)\,\delta(z) + \upsilon T_p\,h_I(z) \tag{3-17}$$

Here h_I is a density function which depends on the pulse shape p and on the density function of the A_t. The quantity $1 - \upsilon T_p$ may be viewed as the probability that no impulse noise is present at the sampling time; it comes from the Poisson assumption about the occurrence times t_t. The result of (3-17) is reasonable for the low-density case $\upsilon T_p \ll 1$, for which there are gaps between successive noise pulses in the impulsive component $I(t)$. Upon

Figure 3.10 Asymptotic Relative Efficiencies of the LO Detectors, Relative to the LC Detector, for Generalized Cauchy Noise

considered nearly Gaussian skewed distributions and have compared the SC and LC detector performances under this condition.

3.3 Mixture Noise and Middleton Class A Noise

Suppose we want to model the behavior of a noise density function which is approximately Gaussian near the origin but has tails which decay at a lower rate than do the Gaussian density tails. An analytically simple model is provided by the mixture density function

$$f(z) = (1 - \epsilon)\eta(z) + \epsilon h(z) \tag{3-15}$$

where ϵ is some small positive constant, η is a Gaussian density function, and h is some other density function with heavier tails. Clearly, f defined by (3-15) is a valid density function as long as ϵ lies in the interval [0,1]. For small enough values of ϵ the

adding an independent Gaussian background noise process to $I(t)$ the first-order density function of the total noise process becomes a convolution of f_I, resulting in the noise density f of (3-15) in which ϵ is now vT_p and h is the convolution of h_I and η.

Middleton [1977, 1979a and 1979b, 1983] has described a canonical model based on a representation of the impulsive component of noise similar to, but more general than, that given by (3-16). For what is called his class A model the pulse widths are wide relative to the receiver filter impulse response so that the receiver filter passes the impulsive components, and physical mechanisms generating the interference can be used to interpret the parameters of the model for the noise. In terms of his two basic parameters (one related to vT_p above, the other being the ratio of noise powers in the impulsive and Gaussian components) Middleton has obtained an expansion of the noise density function f as an infinite weighted sum of Gaussian densities with decreasing weights for Gaussian densities with increasing variances. Specifically, it has been shown that the univariate probability density function of the normalized, unit-variance noise which has a Gaussian component and an independent additive interference component arising from a Poisson mechanism may be approximated as

$$f(z) = \sum_{m=0}^{\infty} \frac{e^{-A} A^m}{m!} \frac{1}{\sqrt{2\pi}\sigma_m} e^{-z^2/(2\sigma_m^2)} \qquad (3\text{-}18)$$

Here the parameter A, called the impulsive index, is like the product vT_p, being the product of an average rate of interfering waveform (pulse) generation and the waveform's mean duration. Thus a small value of A implies highly impulsive interference. The variances σ_m^2 are also related to physical parameters, being given as

$$\sigma_m^2 = \frac{(m/A) + \Gamma'}{1 + \Gamma'} \qquad (3\text{-}19)$$

where Γ' is the ratio of power in the Gaussian component of the noise to the power in the Poisson-mechanism interference. The major appeal of this model is that its parameters have a direct physical interpretation; the class A model has been found to provide very good fits to a variety of noise and interference measurements [Spaulding and Middleton, 1977a, Middleton, 1979a]. The Middleton Class A noise model is actually a model for *narrowband* noise, and it is often the *envelope* of such noise that is to be described statistically. The envelope probability density function for this noise is easily obtained from (3-18) as an infinite mixture of Rayleigh noise densities.

By limiting the sum in (3-18) to the first M terms only, and dividing by the sum of the first M coefficients of the Gaussian densities to maintain normalization, approximations of Middleton's class A model are obtained. It is of significance here that for several cases of empirically fitted noise densities a rather small value for the integer M is found to be sufficient to give excellent approximations to both the noise probability density functions and to the corresponding locally optimum nonlinearities. Such a numerical study has been reported recently by Vastola [1984]. From this observation one concludes that with $M = 2$ terms in (3-18), which does give very good approximations in the instances which have been considered, we end up with the mixture model of (3-15) in which $\epsilon = A/(1 + A)$. Assuming that η and h in (3-15) have zero means and respective variances σ_η^2 and σ_h^2, this also gives us that $\sigma_h^2/\sigma_\eta^2 = (1 + A\Gamma')/A\Gamma'$. In addition, of course, we find that h is also Gaussian here. Thus we find that the two-term mixture model of (3-15) has an additional justification as a special case arising from the class A model. In this particular interpretation the model parameters ϵ and σ_h^2/σ_η^2 can also be given a direct physical interpretation. For typical values of A and Γ, in the ranges (0.01, 0.5) and (0.0001, 0.1), respectively, we find that ϵ is in the range (0.01, 0.33) and σ_h^2/σ_η^2 is in the range (20, 10000). This indicates that the variance of the contaminating h in the mixture model may be much larger than that of the Gaussian component in real situations. We also remark here that Gaussian mixture models in general, of the type described by (3-18), may be thought of as special cases of the type of model mentioned at the end of the preceding section; this is the model of noise with a conditional density function which is Gaussian, conditioned on the variance, which is a random parameter.

Returning to the mixture model of (3-15) in which both η and h have zero means, we find that the variance σ^2 of the mixture noise density is

$$\sigma^2 = (1 - \epsilon)\sigma_\eta^2 + \epsilon\sigma_h^2 \qquad (3\text{-}20)$$

and the LO nonlinearity is

$$g_{LO}(z) = \frac{-(1 - \epsilon)\eta'(z) - \epsilon h'(z)}{(1 - \epsilon)\eta(z) + \epsilon h(z)} \qquad (3\text{-}21)$$

Now we can express this in two alternative ways, using $\eta'(z) = -(z/\sigma_\eta^2)\eta(z)$:

$$g_{LO}(z) = \frac{z/\sigma_\eta^2 - [\epsilon/(1 - \epsilon)]h'(z)/\eta(z)}{1 + [\epsilon/(1 - \epsilon)]h(z)/\eta(z)} \qquad (3\text{-}22)$$

$$g_{LO}(x) = \frac{-h'(z)/h(z) + ((1-\epsilon)/\epsilon)(|z|/\sigma_q^2)\eta(z)/h(z)}{1 + ((1-\epsilon)/\epsilon)\eta(z)/h(z)} \quad (3\text{-}23)$$

Assuming that h' as well as h is bounded, for $\epsilon \ll 1$ we find from (3-22) that for small values of $|z|$ the function g_{LO} is approximately linear in z. If $h'(0)$ is not zero, then for very small values of $|z|$ we will then get a different behavior for g_{LO}, since $g_{LO}(0)$ will then have a value of approximately $-h'(0)/\eta(0)$. Note that $h'(0)$ is strictly speaking $h'(0+)$ or $h'(0-)$. For large values of $|z|$ the behavior of g_{LO} is revealed more easily by (3-23). If the tails of h decay at a lower rate than do those of η, in such a way that $z\eta(z)$ goes to zero faster than do $h(z)$ and $h'(z)$ for $|z| \to \infty$, the second terms in the numerator and denominator in (3-23) become negligible for large values of $|z|$. Thus the limiting behavior of g_{LO} for large $|z|$ will generally be that of the LO characteristic for noise density function h, for heavy-tailed h.

Let h be the double-exponential noise density function of (2-15), in which the parameter a is related to the variance σ_q^2 by $\sigma_q^2 = 2/a^2$. Figure 3.11 shows the characteristic g_{LO} for the Gaussian and double-exponential mixture density with $\epsilon = 0.05$ and $\sigma_q^2/\sigma_n^2 = 10$. The general characteristics of g_{LO} are as we would expect them to be. Note once again that the shape of g_{LO} here is similar to that determined for ELF noise, as given in Figure 3.3.

These results once again indicate that the simple sign correlator detector should perform well in impulsive noise modeled to have the mixture density function, as compared to the linear correlator detector. The ARE of the SC relative to the LO detector for the mixture noise density is given by

$$ARE_{SC,LO} = 4\sigma^2/\eta^2(0)$$

$$= 4[(1-\epsilon)\sigma_q^2 + \epsilon\sigma_n^2]\left[\frac{1-\epsilon}{\sqrt{2\pi\sigma_q^2}} + \epsilon h(0)\right]^2$$

$$= 4[1 - \epsilon + \epsilon\sigma_n^2/\sigma_q^2]\left[\frac{1-\epsilon}{\sqrt{2\pi}} + \sigma_q\epsilon h(0)\right]^2 \quad (3\text{-}24)$$

Note that the factor containing $h(0)$ above has a minimum value of $(1-\epsilon)^2/2\pi$, which is larger than zero for $\epsilon < 1$. It follows then that for $\epsilon > 0$ as well, $ARE_{SC,LO}$ can be made arbitrarily large by making σ_q^2/σ_n^2 sufficiently large. This is true specifically when h is the double-exponential density function. For this contamination noise density function h the $ARE_{SC,LO} = 2$ when $\epsilon = 1$, in which case the SC detector is the LO detector. It is thus interesting that for any ϵ strictly between zero and unity the $ARE_{SC,LO}$ can
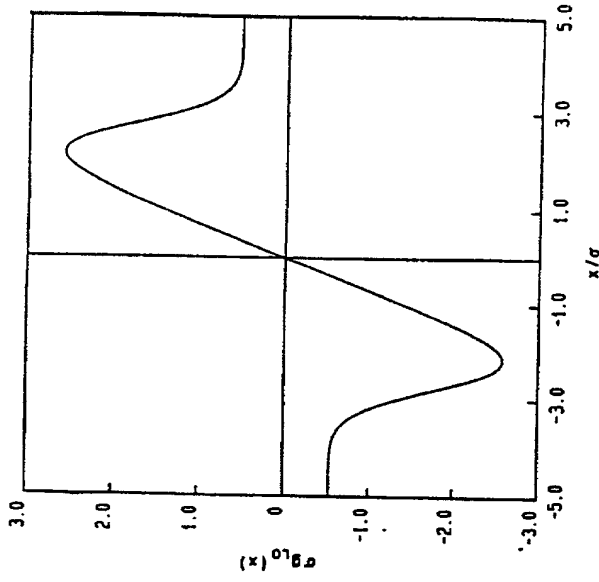
Figure 3.11 LO Nonlinearity for Gaussian and Double-Exponential Noise PDF Mixture ($\epsilon = 0.05$, $\sigma_q^2/\sigma_n^2 = 10$)

become arbitrarily large for such an h. The explanation is that the efficacy of the sign correlator detector is $4f^2(0)$, which for the mixture density always contains a fixed contribution due to η for any σ_n^2. The quantity $4h^2(0)$ by itself decreases to zero for $\sigma_n^2 \to \infty$. The mixture density function f is *not simply scaled* by σ_n^2 for $0 < \epsilon < 1$. Note that the LC detector, on the other hand, has efficacy $1/\sigma^2$, which converges to zero for $\sigma_n^2 \to \infty$.

The factor $[1 - \epsilon + \epsilon\sigma_n^2/\sigma_q^2]$ in (3-24) also has a minimum value of $1 - \epsilon$ which is positive for $\epsilon < 1$. Now $\sigma_q\epsilon h(0) = (\sigma_n/\sigma_q)\epsilon h(0)\sigma_q$, and $h(0)\sigma_q$ has a constant value which we shall assume is not zero. Thus we find that for $0 < \epsilon < 1$ the $ARE_{SC,LO}$ also approaches ∞ as for $\sigma_n^2/\sigma_q^2 \to 0$. For any fixed σ_n^2, the efficacy $1/\sigma^2$ of the LC detector approaches the value $[(1-\epsilon)\sigma_q^2]^{-1}$ for $\sigma_n^2 \to 0$. The efficacy $4f^2(0)$ of the SC detector now contains a contribution due to $\epsilon h(0)$ which, on the other hand, approaches ∞ as σ_n converges to zero, since $h(0)\sigma_n$ is some positive constant. When $\epsilon = 0$ the $ARE_{SC,LO}$ becomes $2/\pi$, its value for Gaussian noise.

We thus see that the SC detector can far outperform the LC detector in highly impulsive noise modeled by the mixture density, for which σ_I^2/σ_g^2 is high, and also in the other extreme case of the mixture model placing a relatively high probability near the origin (Gaussian noise with "quiet" periods). For some numerical evaluations of the $ARE_{SC,LC}$ we refer the reader to the work of Miller and Thomas [1976].

One general conclusion we can make from the specific cases of non-Gaussian noise we have considered here is that with appropriate nonlinear processing of the inputs it is possible to get a substantial advantage over detectors employing the LC test statistic. In all our work we have assumed the availability of independent data components X_i at the detector input. This in turn is obtained under the assumption that the noise is almost "white" relative to the low-pass signal bandwidth. It is important to keep in mind that it is not permissible first to low-pass filter (or narrowband filter) the input process if it contains impulsive non-Gaussian noise. This is not a problem for Gaussian noise, since the very operation of linear-correlation amounts to use of a low-pass filter in any case. For non-Gaussian white noise it is the non-linear limiting type of operation performed by g_{LO} which accounts for the major improvement in performance, through the de-emphasis of large noise values which this implies. The resulting nonlinearly processed data is then effectively low-pass filtered when the GC test statistic is formed. Thus it is permissible to insert an ideal low-pass filter passing the signal frequencies after the zero-memory nonlinearity g_{LO}. But interchanging the order of these functions leads to a different system; the low-pass filter spreads out noise impulses in time, thus very significantly reducing the advantage obtainable with further nonlinear processing. It is for this reason that in practical systems such as that mentioned for ELF noise, pre-whitening or equalization filters are used before hard-limiting of the data to reverse the effect of any low-pass filtering which may have taken place.

One useful indication the results of this and the previous section give us is that detector nonlinearities g which are approximately linear near the origin but which become saturated and tend to limit the influence of large-magnitude observations should give good overall performance for both Gaussian and heavier-tailed impulsive noise density functions. We find, indeed, that such a nonlinearity can give us a minimax robust detection scheme for noise densities modeled by the mixture model (Kassam and Poor [1985]). Other noise models and detector performance studies related to our discussion here may be found in the works of Kurz [1962], Rappaport and Kurz [1966], Klose and Kurz [1969], Chelyshev [1973], Freeman [1981] and Aazhang and Poor [1985, 1986]. Many experimental studies of noise processes arising in various applications have been reported; we refer here to the work of Milne and Ganton [1964] as one example, indicating the highly impulsive nature of underwater noise under certain conditions.

3.4 Adaptive Detection

We have so far always assumed, in obtaining the LO detectors and evaluating their performances, that the noise density function f is completely specified. In this book, in fact, our considerations are primarily confined to such situations and to *fixed* or *non-adaptive* detectors. At least in one special case of incompletely specified noise density an adaptive scheme is quite useful. This is the case where f is known except for its variance σ^2, the noise power. In this case the LO nonlinearity g_{LO} has a known shape, but its input-scaling factor is unknown. More generally, if f is not known, it is unlikely that g_{LO} can be effectively estimated and adaptively updated. But a more reasonable scheme may be possible, in which the test statistic is chosen to be of some simple form with only a few variable parameters. These may then be adaptively set to optimize performance under different noise conditions. We shall focus briefly on this latter topic here, but let us discuss the former first.

Unknown Noise Variance

Let the noise density function be defined as

$$f(x) = \frac{1}{\sigma}\bar{f}(x/\sigma)$$

(3-25)

where $\bar{f}$ is a known unit-variance density and σ^2 is an unknown parameter, the variance of f. The LO characteristic is that given by

$$g_{LO}(x) = -\frac{1}{\sigma}\frac{\bar{f}'(x/\sigma)}{\bar{f}(x/\sigma)}$$
$$= \frac{1}{\sigma}\bar{g}_{LO}(x/\sigma)$$

(3-26)

Note that $\bar{g}_{LO} = -\bar{f}'/\bar{f}$ is now a known function, but g_{LO} is an amplitude as well as input scaled version of $\bar{g}_{LO}$. The amplitude scaling is not essential, and we may write down the LO test statistic as

$$\lambda_{LO}(X) = \sum_{i=1}^{n} s_i \bar{g}_{LO}(X_i/\sigma)$$

(3-27)

Under the noise-only null hypothesis H_1 of (2-13) the density function of X_i/σ is simply $\bar{f}$, so that the null-hypothesis distribution of $\lambda_{LO}(X)$ is exactly known and the detector threshold can be obtained for any false-alarm probability specification. To implement $\lambda_{LO}(X)$ it is necessary to have a good estimate of σ. If the noise power level is non-stationary, periodic updating of the variance is also necessary. Adaptive detection schemes which use input scaling with estimates of σ are quite commonly used in practice, and are commonly refered to as AGC (automatic gain control) schemes. Of course it is important that good estimates of σ be available; otherwise, significant departure from expected performance may result. A related adaptation strategy has been considered by Lu and Eisenstein [1984].

Unknown Noise Density

A useful adaptive detector structure is obtained by allowing only a few variable parameters in the definition of the detector test statistic, which may then be set adaptively to match the noise conditions. One particularly simple structure is suggested by our results of the previous sections, where we saw that the sign correlator detector generally performs quite well for heavy-tailed noise density functions which are typical for impulsive noise. On the other hand, for Gaussian noise the LC detector is optimum and the SC detector performs rather poorly. Now consider the test statistic

$$T_{GC}(X) = \sum_{i=1}^{n} c_i [\gamma X_i + (1-\gamma)\,\text{sgn}(X_i)]$$
$$= \gamma \sum_{i=1}^{n} c_i X_i + (1-\gamma) \sum_{i=1}^{n} c_i \text{sgn}(X_i) \qquad (3\text{-}28)$$

in which γ is a free parameter. The choice $\gamma = 1$ makes this an LC test statistic, and $\gamma = 0$ makes it the SC test statistic. Therefore, if an adaptive scheme can be formulated which allows γ to be picked optimally for different noise density conditions, one can expect the resulting detector to have very good performance over a range of noise conditions.

This type of test statistic was suggested by Modestino [1977], from whose results it follows that the normalized efficacy $\bar{\xi}_\gamma$ of $T_{GC}(X)$ of (3-28) for our detection problem is

$$\bar{\xi}_\gamma = \frac{[\gamma + 2(1-\gamma)f(0)]^2}{\gamma^2 \sigma^2 + 2\gamma(1-\gamma)p_0 + (1-\gamma)^2} \qquad (3\text{-}29)$$

where

$$p_0 = \int_{-\infty}^{\infty} |x| f(x)\, dx \qquad (3\text{-}30)$$

This result for $\bar{\xi}_\gamma$ is quite simple to derive. Modestino [1977] shows that $\bar{\xi}_\gamma$ is maximized for

$$\gamma = \frac{1 - 2p_0 f(0)}{(1-p_0) + 2f(0)(\sigma^2 - p_0)} \qquad (3\text{-}31)$$

and the maximum value of $\bar{\xi}_\gamma$ using this value for γ is

$$\bar{\xi}_{max} = \frac{1 - 4p_0 f(0) + 4\sigma^2 f^2(0)}{\sigma^2 - p_0^2} \qquad (3\text{-}32)$$

For the generalized Gaussian noise densities of (3-1), evaluation of $\bar{\xi}_{max}$ shows that the test statistic of (3-28) with optimum choice of γ has very good asymptotic performance relative to the AO detectors for $1 \leq k \leq 2$, where k is the rate of exponential decay parameter for the generalized Gaussian densities. In fact, the ARE of the optimum detector based on $T_{GC}(X)$ of (3-28) relative to the AO detector is always larger than 0.95 for this range of values.

One possibility in devising an adaptive scheme using $T_{GC}(X)$ of (3-28) is to estimate the parameters p_0, $f(0)$ and σ^2; this requires noise-only training observations. Another possibility is to use a stochastic approximation technique operating on signal-plus-noise observations, which attempts to maximize the output SNR for $T_{GC}(X)$. This approach has been discussed by Modestino [1977]. A detector structure related to the one we have discussed above has been considered in [Czarnecki and Thomas, 1983].

A more general approach is to start with a specified set of nonlinearities $\{g_1(x), g_2(x), ..., g_M(x)\}$, and to consider as possible detector test statistics all linear combinations of these, M functions. An adaptive detector may then be sought which uses coefficients for its linear combination which are optimum, say in the sense of maximizing the detector efficacy, for the prevailing noise probability density function. A useful simplification is obtained if the functions $g_j(x)$, $j = 1,2,...,M$, form an orthonormal set for the allowable noise probability density functions. This means that for allowable f we have

$$\int_{-\infty}^{\infty} g_j(x) g_k(x) f(x)\, dx = \delta_{jk} \qquad (3\text{-}33)$$

where δ_{jk} is the Kronecker delta function. Under this constraint the detector efficacy has a simple expression and the conditions on the linear combination coefficients to maximize efficacy can be found easily (Problem 2.8, Chapter 2). These conditions can also

form the basis for an adaptive detection scheme maximizing performance in different noise environments. A special case is obtained when the orthonormal functions $g_j(x)$ are defined to be non-zero over corresponding sets P_j, $j = 1,2,...,M$, which are *disjoint*. An important example of this is quantization, in which the collection of M functions $\{g_j, j = 1,2,...M\}$ together with the set of linear combination weighting coefficients define a quantizer characteristic. We shall discuss these ideas further at the end of the next chapter.

While it is possible, then, to consider some simplified schemes for adaptive detection in unknown and nonstationary noise environments, the implementation of an efficient adaptive scheme necessarily adds, often significantly, to the complexity of the detector. Furthermore, there are considerations such as of sample sizes required to achieve convergence which have to be taken into account in designing a useful scheme of this type.

PROBLEMS

Problem 3.1

An alternative parameterization of the generalized Gaussian pdf's of (3-1) is in terms of k and $a = A(k)$, giving

$$f_k(z) = \frac{k}{2a\Gamma(1/k)} e^{-|z/a|^k}$$

Note that $A(k)$ was defined in (3-2) in terms of σ^2, the variance.

(a) Show that for fixed value of a the pdf f_k converges for $k \to \infty$ to the *uniform* pdf on $[-a, a]$.

(b) Find the limit to which $ARE_{SC,LC}$ of (3-8) converges, as $k \to \infty$. Verify that this agrees with the result of a direct calculation of $ARE_{SC,LC}$ for the uniform pdf.

Problem 3.2

Let V be a Gaussian random variable with mean zero and variance unity. Let Z be an independent Rayleigh random variable with pdf

$$f_Z(z) = \frac{z}{a^2} e^{-z^2/2a^2}, \quad z \geq 0$$

Find the pdf of $X = ZV$. (This is a model for a conditionally Gaussian noise sample with a random power level.)

Problem 3.3

The symmetric, unimodal *logistic* pdf is defined as

$$f(z) = \frac{1}{a} \frac{e^{-z/a}}{[1 + e^{-z/a}]^2}, \quad -\infty < z < \infty$$

(a) Prove that its variance is $a^2 \pi^2/3$. (Use a series expansion.) Plot on the same axes the zero-mean unit-variance Gaussian, double-exponential, and logistic pdf's. (The logistic pdf is useful as a non-Gaussian pdf having exponential tail behavior and smooth behavior at the origin.)

(b) Show that g_{LO} for this pdf is given by

$$g_{LO}(z) = \frac{1}{a} [2F(z) - 1]$$

where F is the cumulative distribution function corresponding to the pdf f above. Sketch $g_{LO}(z)$.

(c) Find $ARE_{LC,LO}$ and $ARE_{SC,LO}$ in testing H_1 versus K_1 of (2-13) and (2-14) when f is the logistic pdf.

Problem 3.4

Obtain $ARE_{SC,LC}$ for generalized Cauchy noise in (2-13) and (2-14), and sketch the result as a function of k for a small, an intermediate, and a large value of v.

Problem 3.5

In the mixture model of (3-15) let η and h both be zero-mean Gaussian pdf's, with η having unit variance. For $\sigma_h^2 = 100$ and $\epsilon = 0.05$ plot the LO nonlinearity using log scales for both axes.

Problem 3.6

Let f be the mixture noise pdf of (3-15) in which η is the zero-mean, unit-variance Gaussian pdf and h is a bounded symmetric pdf. Consider the GC detector based on the soft-limiter function defined in Problem 2.5 of Chapter 2; we will denote this soft-limiter function as l_c.

(a) For fixed ϵ, find the mixture proportion ϵ and the pdf h (in terms of η and c) for which the resulting mixture pdf $f = f^*$ has l_ϵ as the LO nonlinearity g_{LO}.

(b) Consider the class of mixture pdf's obtained with different bounded symmetric h in the mixture model with η fixed as above and with ϵ fixed in terms of ϵ and η as above. Show that the efficacy of the GC detector based on l_ϵ for H_1 versus K_1, is a minimum for the pdf f^* in this class. (This is the essence of the min-max robustness property of the soft limiter.)

```
/* this program takes in a raw 8 bit image, with x and y dimensions as inputs also,
   and encodes that image with a specified number, spitting the encoded image out */

/* the current version, started just after sign_it.sav3, now adds the three parameters
   of dc, root of the gray value, and a constant times the local gradient */


#include "pinecone.h"
#include "image.h"

#define NOISE_MAX 10000000
#define BITS 16
#define HALF_MAX 32768
#define MAX 65536
#define MAX1 65535
#define GV_POWER 0.4
#define GRAD_THRESHOLD 50.0

#define DISPLAY_IT 1

unsigned char *img;
long bit[BITS];
long gid_gv,gid_grad;
long xdim,ydim;
float *gv,*grad;
Colorindex *disp;
Colorindex *scale_disp;
Colorindex *pdisp;

int helplines = 2;
char    *help[] =
{
        "usage: sign_it filename [xdim(default 510) ydim(default 256)]\n",
        "\nMust input both xdim and ydim if not using the defaults "
};

int fill_bits(int code_number){
long i,tmp;

printf("The binary value ");

tmp = HALF_MAX;
for(i=0;i<16;i++){
        if(code_number>=tmp){
                printf("1");
                code_number-=tmp;
```

Appendix B

```c
                bit[i]=1;
        }
        else {
                printf("0");
                bit[i]=0;
        }
        tmp/=2;
}
printf(" will be wallpapered across the image\n\n");
return(0);
}

int load_scale_disp(void){
long i,j;

        pdisp = scale_disp+20;
        for(j=0;j<15;j++){
                for(i=1;i<20;i++){
                        *pdisp = (unsigned char)255;
                        pdisp += 20;
                }
                pdisp += 20;
        }
        pdisp = scale_disp+20+(35*400);
        for(j=0;j<15;j++){
                for(i=1;i<20;i++){
                        *pdisp = (unsigned char)255;
                        pdisp += 20;
                }
                pdisp += 20;
        }
return(0);
}

int draw_scale(float value,long which){
long i;
float ftemp;

value *=5;

        memcpy(disp,scale_disp,40000);
        ftemp = 20.0 * value;
        if(ftemp > 399.0)ftemp=399.0;
        pdisp = &disp[15*400 + (int)ftemp];
        for(i=0;i<20;i++){
                *pdisp = (unsigned char)255;
```

2

```c
                pdisp += 400;
        }
        winset(which);
        rectwrite(0,0,399,49,disp);

return(0);
}

int calculate_gv(){
unsigned char *pimg;
long i;
float *pgv;

        pimg = img;
        pgv = gv;
        for(i=0;i<(ydim*xdim);i++){
                *(pgv++) = powf( (float)*(pimg++) , GV_POWER );
        }

return(0);
}

int calculate_grad(){
unsigned char *pimg1,*pimg2,*pimg;
long i,j;
float avg,*pgrad;

        pimg = &img[1+xdim];
        pimg1 = &img[1];
        pimg2 = &img[1+2*xdim];
          pgrad = grad;
        for(i=0;i<(xdim*ydim);i++)*(pgrad++) = 0.0;
          pgrad = &grad[1+xdim];
        for(i=1;i<(ydim-1);i++){
                for(j=1;j<(xdim-1);j++){
avg = (float)*(pimg1-1)+ (float)*(pimg1++)+ (float)*pimg1 + (float)*(pimg-
1)+(float)*(pimg+1)+(float)*(pimg2-1)+(float)*(pimg2++) + (float)*pimg2;
                        avg/=8.0;
                        *(pgrad++) = (float)*(pimg++) - avg;
                }
                pimg+=2; pimg1+=2; pimg2+=2; pgrad+=2;
        }

return(0);
}
```

```c
main( argc, argv )
int     argc ;
char    *argv[] ;
{
        long i,j,code_number,go=1,cur_bit,button;
        long temp,increment,last_middle,count_thresh = 0;
        long int_total[BITS];
        unsigned char *pimg,tmp,*pimg1,*img_out,*noise,*pnoise;
        char string[80], outfile[80];
        FILE *inf;
        float ftemp,current_scale = 1.0,rms,avg;
        float current_gv = 0.0035, current_grad = 0.0027;
        float f_inc,mult;
        float *pgrad,*pgv;
        long gid_img,gid_img1,gid_signed,which;

        if(argc<2 || argc ==3 || argc>4) {
                for(j=0;j<helplines;j++)fprintf( stderr, "%s", help[j]) ;
                exit( 1 ) ;
        }
        else if(argc ==2){
                xdim = 324;
                ydim = 414;
        }
        else {
                xdim = atoi( argv[2] );
                ydim = atoi( argv[3] );
        }

if(DISPLAY_IT) disp = calloc(xdim*ydim, sizeof(Colorindex) ) ;
else disp = calloc(40000, sizeof(Colorindex) ) ;
        scale_disp = calloc(40000, sizeof(Colorindex) ) ;
        img = calloc(xdim*ydim, sizeof(unsigned char) ) ;
        img_out = calloc(xdim*ydim, sizeof(unsigned char) ) ;
        noise = calloc(xdim*ydim, sizeof(unsigned char) ) ;
        gv = calloc(xdim*ydim, sizeof(float) ) ;
        grad = calloc(xdim*ydim, sizeof(float) ) ;
        if( !disp || !scale_disp || !img || !img_out || !noise || !gv || !grad){
                fprintf( stderr, "sign_it : can not allocate space\n" ) ;
                exit( 1 ) ;
        }

        /* read in binary data into array */
        inf = fopen(argv[1],"r");
        if(!inf) {
```

```c
            fprintf(stderr,"sign_it: can't open %s\n",argv[1]);
            exit(1);
}
fread(img,sizeof(unsigned char),xdim*ydim,inf);
fclose(inf);

/* flip it */
pimg = img;
pimg1 = &img[xdim*(ydim-1)];
for(i=0;i<(ydim/2);i++){
        for(j=0;j<xdim;j++){
                tmp = *pimg;
                *(pimg++) = *pimg1;
                *(pimg1++) = tmp;
        }
        pimg1 -= (2*xdim);
}

/* open and read in master noise frame */
inf = fopen("master_noise","r");
if(!inf) {
        fprintf(stderr,"sign_it: can't open %s\n","master_noise");
        exit(1);
}
if((xdim*ydim) > NOISE_MAX ){
        fprintf(stderr,"sign_it: image size larger than master key ");
        exit(1);
}
fread(noise,sizeof(unsigned char),xdim*ydim,inf);
fclose(inf);

/* calculate avg then rms noise of master_noise image */
pnoise = noise;
avg = 0.0;
for(i=0;i<(ydim*xdim);i++){
                ftemp = (float)*(pnoise++);
                avg += ftemp;
}
avg /= (float)(xdim*ydim);
printf("AVG noise value of master noise image = %f \n\n",avg);
pnoise = noise;
rms = 0.0;
for(i=0;i<(ydim*xdim);i++){
                ftemp = (float)*(pnoise++) - avg ;
                ftemp *= ftemp;
                rms += ftemp;
```

5

```
        }
        rms /= (xdim*ydim);
        rms = (float)sqrt((double)rms);
        printf("RMS noise value of master noise image = %f \n\n",rms);

        /* calculate gv */
        calculate_gv(img,gv,xdim,ydim);

        /* calculate grad */
        calculate_grad(img,grad,xdim,ydim);

        /* user input */
        code_number = -1;
        while(code_number < 0 || code_number > MAX1){
                printf("Type in a number between 0 and %ld: ",MAX1);
                scanf("%d",&code_number);
        }
        fill_bits(code_number);

if(DISPLAY_IT){
        foreground();
        prefsize(xdim,ydim);
        gid_img = winopen("Original?");
        gflush();

        pdisp = disp;
        pimg = img;
        for(i=0;i<(ydim*xdim);i++){
                        *(pdisp++) = (Colorindex)*(pimg++);
        }
        rectwrite(0,0,xdim-1,ydim-1,disp);

        prefsize(xdim,ydim);
        gid_img1 = winopen("Orignal?");
        rectwrite(0,0,xdim-1,ydim-1,disp);

        prefsize(xdim,ydim);
        gid_signed = winopen("Orignal?");

        load_scale_disp();
        prefsize(400,50);
        gid_gv = winopen("root GV rms noise");
        prefsize(400,50);
        gid_grad = winopen("GRAD rms noise");

}
```

```c
/* main visual feedback loop */
last_middle = 0;
while(go){
        pdisp = disp;
        pimg = img;
        pimg1 = img_out;
        pnoise = noise;
        pgv = gv;
        pgrad = grad;
        for(i=0;i<(ydim*xdim);i++){
                cur_bit = i%BITS;
                mult = current_gv * *(pgv++) + current_grad * *(pgrad++);
                if( (float)*pnoise > avg ){
                        f_inc = ((float)*(pnoise++) - avg );
                        increment = (int)(f_inc * mult);
                }
                else {
                        f_inc = (avg - (float)*(pnoise++) );
                        increment = (int)(f_inc * mult);
                        increment = -increment;
                }
                if( fabs( (double) *(pgrad - 1)) > GRAD_THRESHOLD ){
                        increment = 0;
                }
                if(bit[cur_bit]){
                        temp =  (int)*(pimg) + increment;
                }
                else{
                        temp =  (int)*(pimg) - increment;
                }
                if( *(pimg++) == 0 )tmp=0;
                else if(temp < 0)tmp=0;
                else if(temp > 255)tmp=255;
                else tmp= (unsigned char)temp;
                if(DISPLAY_IT) *(pdisp++) = (Colorindex)tmp;
                *(pimg1++) = tmp;
        }
if(DISPLAY_IT){
        winset(gid_signed);
        rectwrite(0,0,xdim-1,ydim-1,disp);
        pdisp = disp;
        pimg = img;
        for(i=0;i<(ydim*xdim);i++){
                *(pdisp++) = (Colorindex)*(pimg++);
        }
```

7

```
                winset(gid_img);
                rectwrite(0,0,xdim-1,ydim-1,disp);
                winset(gid_img1);
                rectwrite(0,0,xdim-1,ydim-1,disp);

                draw_scale(rms*current_gv,gid_gv);
                draw_scale(rms*current_grad,gid_grad);
        }

                button=0;
                while(!button){
                        current_scale=1.0;
                        if( getbutton(LEFTMOUSE) ){
                                current_scale *= 1.15;
                                button = 1;
                                last_middle = 0;
                        }
                        if( getbutton(RIGHTMOUSE) ){
                                current_scale *= 0.85;
                                button = 1;
                                last_middle = 0;
                        }
                        if( getbutton(MIDDLEMOUSE) ){
                                while( getbutton(MIDDLEMOUSE) );
                                if( last_middle ){
                                        button = 1;
                                        go=0;
                                }
                                else {
                                        if(which == gid_gv)which=gid_grad;
                                        else which=gid_gv;
                                        last_middle = 1;
                                }
                        }
                }
                if( go ){
                        if( which ==  gid_gv )current_gv *= current_scale;
                        else current_grad *= current_scale;
                }
        }

/* re-calculate and write out master_key */
pnoise = noise;
pimg = img;  /* use img as a temporary buffer */
pgv = gv;
pgrad = grad;
```

8

```
count_thresh = 0;
for(i=0;i<(ydim*xdim);i++){
        mult = current_gv * *(pgv++) + current_grad * *(pgrad++);
        if( (float)*pnoise > avg ){
                f_inc = ((float)*(pnoise++) - avg );
                increment = (int)(f_inc * mult);
        }
        else {
                f_inc = (avg - (float)*(pnoise++) );
                increment = (int)(f_inc * mult);
                increment = -increment;
        }
        if( fabs( (double) *(pgrad - 1)) > GRAD_THRESHOLD ){
                count_thresh++;
                increment = 0;
        }
        *(pimg++) = (unsigned char)128 + (unsigned char)increment;
}

/* find mean value of the integer based noise,
        for each bit plane, then remove it, and store to dnoise */
for(i=0;i<BITS;i++)int_total[i] = 0;
pimg = img;
for(i=0;i<(ydim*xdim);i++){
        int_total[i%BITS] += (long) (*(pimg++) - 128 );
}
rms = 0.0;
for(j=0;j<BITS;j++){
        if(int_total[j] > 0){
                for(i=0;i<int_total[j];i++){
                        temp = (long)rand();
                        temp *= BITS;
                        temp += j;
                        if( (temp <(xdim * ydim)) && (img[temp] > 128)
)img[temp]--;
                        else i--;
                }
        }
        else if(int_total[j] < 0){
                int_total[j] = - int_total[j];
                for(i=0;i<int_total[j];i++){
                        temp = (long)rand();
                        temp *= BITS;
                        temp += j;
                        if( temp <(xdim * ydim) && (img[temp] < 128)
)img[temp]++;
```

9

```c
                        else i--;
                }
        }
}
rms = 0.0;
pimg = img;
for(i=0;i<(ydim*xdim);i++){
        temp = (long) *(pimg++) - 128;
        rms += (float)(temp*temp);
}
rms /= (xdim * ydim);
rms = (float)sqrt((double)rms);
printf("\nFINAL RMS noise value of master key = %f \n\n",rms);
printf("count threshold hits = %d \n\n",count_thresh);
sprintf(outfile,"master_key");
inf = fopen(outfile,"w");
if(!inf) {
        fprintf(stderr,"sign_it: can't open %s\n",outfile);
        exit(1);
}
fwrite(img,sizeof(unsigned char),xdim*ydim,inf);
fclose(inf);

/* now re-sign output image because of slight changes to master key */


/* flip output image */
pimg = img_out;
pimg1 = &img_out[xdim*(ydim-1)];
for(i=0;i<(ydim/2);i++){
        for(j=0;j<xdim;j++){
                tmp = *pimg;
                *(pimg++) = *pimg1;
                *(pimg1++) = tmp;
        }
        pimg1 -= (2*xdim);
}

/* write out signed image */
sprintf(outfile,"%s.signed",argv[1]);
inf = fopen(outfile,"w");
if(!inf) {
        fprintf(stderr,"sign_it: can't open %s\n",outfile);
        exit(1);
}
fwrite(img_out,sizeof(unsigned char),xdim*ydim,inf);
```

10

```
        fclose(inf);


        /* free and clean up */
if(DISPLAY_IT) gexit();
        free(img);
        free(img_out);
        free(disp);
        free(scale_disp);
        free(noise);
        free(gv);
        free(grad);

}
```

/* this program takes in a raw 8 bit image which is in general a sub-section of another original image. The orginal image will have already been signed. This program needs to know what offest and extent that the sub-section has realtive to the dimensions of the original. It also needs to have the master_noise file present keyed to the original extent of the image also */

/* this one now uses the dot product approach to remove the innate bias from the original */

```c
#include "pinecone.h"
#include "../../sgi_tools/image.h"

#define BITS 16
#define VALUE 32768
#define INCREMENT 0.5

int bit[BITS];
long dim,bits,fftdim,fft_size;
float *ar,*ai;
double wr[10000],wi[10000];

int helplines = 7;
char    *help[] =
{
        "usage: recognize filename original_file orig_xdim orig_ydim x_offset y_offset
patch_xdim patch_ydim\n",
        "\n Program requires that the file 'master_key' exists in current directory",
        "\n filename is image in 8-bit raw form which is in general sub-section of
original image",
        "\n original_file is original image in 8-bit raw form ",
        "\n orig_xdim and orig_ydim is the pixel extent of the original image",
        "\n x_offset and y_offset is the offset in pixels of the upper left edge of the
sub-section rel. to original",
        "\n patch_xdim and patch_ydim is the pixel extent of the image patch being
recognized"
};


int general_purpose_filter(float *image, long xdim, long ydim, long which){
long i,j;
float *par,*pai,*pan,*pan_im,*pimage;
float k_value,zero_point,ii,jj,ftemp;
float std_dev=0.0;
```

Appendix C

```c
memset(ar,0, fft_size * sizeof(float) );
memset(ai,0, fft_size * sizeof(float) );

/* copy image buffer into ar */
par = ar;
pimage = image;
for(i=0;i<ydim;i++){
        for(j=0;j<xdim;j++){
                *(par++) = *(pimage++);
        }
        par += (fftdim - xdim);
}

/* real 2d fft, in place */
realfft2d(ar,ai,bits,0,wr,wi);

/* if which == 0 or 1, find the standard deviation in frequency space */
if(which == 0 || which == 1){
        par = ar; pai = ai;
        pan = &ar[fftdim-1];pan_im=&ai[fftdim-1];
        std_dev = 0.0;
        for(i=0;i<(1+fftdim/2);i++){
                ii = (float)i / (float)(fftdim/2);
                ii *= ii;
                for(j=0;j<(1+fftdim/2);j++){
                        jj = (float)j / (float)(fftdim/2);
                        jj = (float)sqrt((double)(ii+jj*jj));

                        ftemp = *par * *par + *pai * *pai;
                        std_dev += jj*ftemp;
                        par++; pai++;
                        if(j!=0 && j!=(fftdim/2)){
                                ftemp = *pan * *pan + *pan_im * *pan_im;
                                std_dev += jj*ftemp;
                                pan--; pan_im--;
                        }

                }
                par += (fftdim/2 -1); pai += (fftdim/2 -1);
                pan += (fftdim +fftdim/2 -1); pan_im += (fftdim+fftdim/2 -1);
        }
        printf("\n Std deviation found = %e ",std_dev);
}


/* now filter it in the case of which == 1 */
```

2

```c
if(which == 3){
        k_value = 1.0 / powf(zero_point,3.0);
        if(zero_point > 0.99 ) zero_point = 0.99;
        par = ar;pai=ai;
        pan = &ar[fftdim-1];pan_im=&ai[fftdim-1];
        for(i=0;i<(1+fftdim/2);i++){
                ii = (float)i / (float)(fftdim/2);
                ii *= ii;
                for(j=0;j<(1+fftdim/2);j++){
                        jj = (float)j / (float)(fftdim/2);
                        jj = (float)sqrt((double)(ii+jj*jj));

                        if( jj < zero_point ){
                                jj = 1.0 - (k_value*jj*jj*jj);
                                *(par++) *= jj;
                                *(pai++) *= jj;
                                if(j!=0 && j!=(fftdim/2)){
                                        *(pan--) *= jj;
                                        *(pan_im--) *= jj;
                                }
                        }
                        else {
                                *(par++) = 0.0;
                                *(pai++) = 0.0;
                                if(j!=0 && j!=(fftdim/2)){
                                        *(pan--) = 0.0;
                                        *(pan_im--) = 0.0;
                                }
                        }
                }
                par += (fftdim/2 -1); pai += (fftdim/2 -1);
                pan += (fftdim +fftdim/2 -1); pan_im += (fftdim+fftdim/2 -1);
        } .
}
```

/* and in the case of which == 2, then use the same filter used in the case 1 */

/* in the case of 1 and 2, inverse fft and copy back into the original pointer array */

```
if( which == 3  || which ==2){
        realfft2d( ar, ai, bits, 1, wr, wi) ;

        par = ar;
        pimage = image;
        for(i=0;i<ydim;i++){
                for(j=0;j<xdim;j++){
                        *(pimage++) = *(par++);
                }
                par += (fftdim - xdim);
        }
}

return(1);
}




double dot_product(float *img1,float *img2,long size){
long i;
float *pimg1,*pimg2;
double result = 0.0;

pimg1 = img1;
pimg2 = img2;
        for(i=0;i<size;i++) result += ( (double) *(pimg1++)  * (double) *(pimg2++) );

return(result);
}

double dot_squared_product(float *img1,float *img2,long size){
long i;
float *pimg1,*pimg2;
double result = 0.0;
double dtemp;

pimg1 = img1;
pimg2 = img2;
        for(i=0;i<size;i++){
                dtemp = (double)*(pimg1++);
                if(dtemp < 0.0){
                        dtemp*=dtemp;
                        dtemp = -dtemp;
                }
                else dtemp *= dtemp;
```

```c
                result += ( dtemp  * (double) *(pimg2++) );
        }


return(result);
}

double dot_root_product(float *img1,float *img2,long size){
long i;
float *pimg1,*pimg2;
double result = 0.0;
double dtemp;

pimg1 = img1;
pimg2 = img2;
        for(i=0;i<size;i++){
                dtemp = (double)*(pimg1++);
                if(dtemp <0){
                        dtemp = sqrt(-dtemp);
                        dtemp = -dtemp;
                }
                else dtemp = sqrt(dtemp);
                result += ( dtemp  * (double)*(pimg2++) );
        }
return(result);
}

double dot_rootroot_product(float *img1,float *img2,long size){
long i;
float *pimg1,*pimg2;
double result = 0.0;
double dtemp;

pimg1 = img1;
pimg2 = img2;
        for(i=0;i<size;i++){
                dtemp = (double)*(pimg1++);
                if(dtemp <0){
                        dtemp = sqrt(-dtemp);
                        dtemp = sqrt(dtemp);
                        dtemp = -dtemp;
                }
                else {
                        dtemp = sqrt(dtemp);
                        dtemp = sqrt(dtemp);
                }
```

```c
                result += ( dtemp  * (double)*(pimg2++) );
        }
return(result);
}


main( argc, argv )
int     argc ;
char    *argv[] ;
{
        long i,j,k,xdim,ydim,temp,patch_xdim,patch_ydim,xoffset,yoffset,index;
        long value = VALUE,code_number = 0,go;
        unsigned char tmp,*noise,*pnoise,*original,*poriginal,*p1;
        unsigned char *img,*pimg,*pimg1,stmp;
        char string[80], outfile[80];
        FILE *inf;
        float *fnoise,*pfnoise,total,*fimg,*pfimg,*foriginal,*pforiginal;
        float *fnoise_tmp,*pfnoise_tmp;
        double base_dot,dot,img_mag,original_mag,noise_mag,rms;
        double square_dot,root_dot,rootroot_dot;
        double base_square_dot,base_root_dot,dtemp;
        double img_mag_root,img_mag_square,img_mag_rootroot;
        double dot1,root_dot1,root_dot2;

        if(argc != 9) {
                for(j=0;j<helplines;j++)fprintf( stderr, "%s", help[j]) ;
                exit( 1 ) ;
        }
        xdim = atoi( argv[3] );
        ydim = atoi( argv[4] );
        xoffset = atoi( argv[5] );
        yoffset = atoi( argv[6] );
        patch_xdim = atoi( argv[7] );
        patch_ydim = atoi( argv[8] );

        /* find the next power of two equal to or higher than patch */
        if(patch_xdim > patch_ydim)dim = patch_xdim;
        else dim = patch_ydim;
        fftdim = 1; go = 1; bits = 0;
        while( go ){
                if( dim > fftdim ){
                        fftdim*=2;
                        bits++;
                }
                else go = 0;
        }
```

```
        if(bits > 12){
                fprintf( stderr, "recognize : sorry, this particular program only accepts
4K images and less\n" ) ;
                exit( 1 ) ;
        }
        fft_size = fftdim * fftdim;

        img = calloc(patch_xdim*patch_ydim, sizeof(unsigned char) ) ;
        fimg = calloc(patch_xdim*patch_ydim, sizeof(float) ) ;
        original = calloc(xdim*ydim, sizeof(unsigned char) ) ;
        foriginal = calloc(patch_xdim*patch_ydim, sizeof(float) ) ;
        noise = calloc(xdim*ydim, sizeof(unsigned char) ) ;
        fnoise = calloc(patch_xdim*patch_ydim, sizeof(float) ) ;
        fnoise_tmp = calloc(patch_xdim*patch_ydim, sizeof(float) ) ;
        ar = calloc(fft_size, sizeof(float) ) ;
        ai = calloc(fft_size, sizeof(float) ) ;
        if( !img || !noise || !original || !fnoise || !fnoise_tmp || !fimg || !foriginal || !ar ||
!ai){
                fprintf( stderr, "sign_it : can not allocate space\n" ) ;
                exit( 1 ) ;
        }


        /* read in Suspected Patch into img */
        inf = fopen(argv[1],"r");
        if(!inf) {
                fprintf(stderr,"sign_it: can't open %s\n",argv[1]);
                exit(1);
        }
        fread( img,sizeof(unsigned char),patch_xdim*patch_ydim,inf);
        fclose(inf);

        /* flip it */
        pimg = img;
        p1 = &img[patch_xdim*(patch_ydim-1)];
        for(i=0;i<(patch_ydim/2);i++){
                for(j=0;j<patch_xdim;j++){
                        tmp = *pimg;
                        *(pimg++) = *p1;
                        *(p1++) = tmp;
                }
                p1 -= (2*patch_xdim);
        }

        /* copy it into float array, removing mean value */
        pimg = img;
```

```
total = 0.0;
for(i=0;i<(patch_ydim*patch_xdim);i++)total += (float) *(pimg++);
total /= (float)(patch_xdim*patch_ydim);
pimg = img;
pfimg = fimg;
for(i=0;i<(patch_ydim*patch_xdim);i++) *(pfimg++) = (float) *(pimg++) - total;
pfimg = fimg;
img_mag = 0.0;
for(i=0;i<(patch_ydim*patch_xdim);i++){
        dtemp = (double)*(pfimg++);
        dtemp *= dtemp;
        img_mag += dtemp;
}
img_mag = sqrt(img_mag);
printf("img_mag = %lf ",img_mag);

/* now find its std deviation in frequency space using the general purpose
filter function */
/*
        general_purpose_filter(fimg,patch_xdim,patch_ydim,0);
*/

/* read in Original Unsigned Image */
inf = fopen(argv[2],"r");
if(!inf) {
        fprintf(stderr,"sign_it: can't open %s\n",argv[2]);
        exit(1);
}
fread(original,sizeof(unsigned char),xdim*ydim,inf);
fclose(inf);

/* create original patch */
index = yoffset * xdim + xoffset;
poriginal = original;
p1 = &original[index];
for(j=0;j<patch_ydim;j++){
        for(k=0;k<patch_xdim;k++){
                *(poriginal++) = *(p1++);
        }
        p1 += (xdim - patch_xdim);
}

/* flip it */
poriginal = original;
p1 = &original[patch_xdim*(patch_ydim-1)];
for(i=0;i<(patch_ydim/2);i++){
```

8

```c
for(j=0;j<patch_xdim;j++){
        tmp = *poriginal;
        *(poriginal++) = *p1;
        *(p1++) = tmp;
}
p1 -= (2*patch_xdim);
}

/* copy it into float array, removing mean */
poriginal = original;
total = 0.0;
for(i=0;i<(patch_ydim*patch_xdim);i++)total += (float) *(poriginal++);
total /= (float)(patch_xdim*patch_ydim);
poriginal = original;
pforiginal = foriginal;
for(i=0;i<(patch_ydim*patch_xdim);i++) *(pforiginal++) = (float) *(poriginal++) -
total;

/* now find its std deviation in frequency space, then filter it to roughly match
the suspect patch */
/*
        general_purpose_filter(foriginal,patch_xdim,patch_ydim,1);
*/

pforiginal = foriginal;
original_mag = 0.0;
for(i=0;i<(patch_ydim*patch_xdim);i++){
        dtemp = (double)*(pforiginal++);
        dtemp *= dtemp;
        original_mag += dtemp;
}
original_mag = sqrt(original_mag);
printf("original_mag = %lf \n",original_mag);




/* now remove the original image from img by using the dot product, or
whatever */
        dot = dot_product(foriginal,fimg,patch_xdim*patch_ydim);
printf(" initial dot = %le   ",dot/(original_mag*img_mag));
        dot /= (original_mag * original_mag);
printf("dot found = %le ",dot);
        pforiginal = foriginal;
        pfimg = fimg;
        img_mag = 0.0;
        for(i=0;i<(patch_ydim*patch_xdim);i++){
                *pfimg -= (dot * *(pforiginal++));
```

9

```
/* now re-calculate the magnitudes */
if( *pfimg < 0.0 ) dtemp = sqrt( -(double)*pfimg );
else dtemp = sqrt( (double)*pfimg );
img_mag_rootroot += dtemp;
dtemp = (double)*(pfimg++);
if(dtemp < 0.0) img_mag_root -= dtemp;
else img_mag_root += dtemp;
dtemp *= dtemp;
img_mag += dtemp;
dtemp *= dtemp;
img_mag_square += dtemp;
}
img_mag = sqrt(img_mag);
img_mag_rootroot = sqrt(img_mag_rootroot);
img_mag_root = sqrt(img_mag_root);
img_mag_square = sqrt(img_mag_square);
dot = dot_product(foriginal,fimg,patch_xdim*patch_ydim);
printf(" new dot = %le \n",dot/(original_mag*img_mag));



/* open and read in master noise frame */
inf = fopen("master_key","r");
if(!inf) {
        fprintf(stderr,"sign_it: can't open %s\n","master_key");
        exit(1);
}
fread(noise,sizeof(unsigned char),xdim*ydim,inf);
fclose(inf);
yoffset = ydim - (yoffset + patch_ydim);  /* needed for flipping */



/* main recoginition loop */

rms = 0.0;
for(i=0;i<BITS;i++){

        /* create current bit plane noise patch, fnoise */
        /* THE MEAN OF THESE MASTER NOISE PLANES SHOULD BE
ZERO!!! FROM THE SIGNING PROCESS */
        index = yoffset * xdim + xoffset;
        pnoise = &noise[index];
        pfnoise = fnoise;
        for(j=0;j<patch_ydim;j++){
```

10

```c
        for(k=0;k<patch_xdim;k++){
                if( (index%BITS) == i ){
                        *pfnoise = (float) (*pnoise - 128);
                        rms += (double)*pfnoise * (double)*pfnoise;
                }
                else *pfnoise = 0.0;
                pfnoise++;
                pnoise++;
                index++;
        }
        pnoise += (xdim - patch_xdim);
        index += (xdim - patch_xdim);
}

/* now filter that patch using the same gaussian used on the original */

general_purpose_filter(fnoise,patch_xdim,patch_ydim,2);

pfnoise = fnoise;
noise_mag = 0.0;
for(j=0;j<(patch_ydim*patch_xdim);j++){
        noise_mag += (double) *(pfnoise) * (double) *(pfnoise);
        pfnoise++;
}
noise_mag = sqrt(noise_mag);

/* find dot product between suspect patch and filtered tmp noise */
dot = dot_product(fnoise,fimg,patch_xdim*patch_ydim);
dot /= (noise_mag * img_mag);

printf("dot=%le ",dot);

/* new expiriment: re-weight the noise then do the root dot */
        pfimg = fimg;
        pfnoise = fnoise;
        noise_mag = 0.0;
        for(j=0;j<patch_ydim;j++){
                for(k=0;k<patch_xdim;k++){
                        dtemp = (double)*pfimg;
                        if( fabs( dtemp ) < 1.0 );
                        else {
                                if( dtemp >= 0.0){
                                        *pfnoise /= (float)sqrt( dtemp );
                                }
                                else {
                                        *pfnoise /= (float)sqrt( -dtemp );
```

11

```c
                                }
                        }
                        noise_mag += (double) *(pfnoise) * (double) *(pfnoise);
                        pfnoise++; pfimg++;
                }
        }
        noise_mag = sqrt(noise_mag);
        root_dot = dot_root_product(fimg,fnoise,patch_xdim*patch_ydim);
        root_dot /= (noise_mag * img_mag_root);
        printf("   new root = %le ",root_dot);

        pfimg = fimg;
        pfnoise = fnoise;
        pfnoise_tmp = fnoise_tmp;
        noise_mag = 0.0;
        for(j=0;j<patch_ydim;j++){
                for(k=0;k<patch_xdim;k++){
                        dtemp = (double)*pfimg;
                        if( fabs( dtemp ) < 1.0 )*pfnoise_tmp = *pfnoise;
                        else {
                                if( dtemp >= 0.0){
                                        *pfnoise_tmp = *pfnoise / (float)sqrt(
dtemp );

                                }
                                else {
                                        *pfnoise_tmp = *pfnoise / (float)sqrt( -
dtemp );

                                }
                        }
                        noise_mag += (double) *(pfnoise_tmp) * (double)
*(pfnoise_tmp);

                        pfnoise++; pfimg++;pfnoise_tmp++;
                }
        }
        noise_mag = sqrt(noise_mag);
        root_dot = dot_root_product(fimg,fnoise_tmp,patch_xdim*patch_ydim);
        root_dot /= (noise_mag * img_mag_root);
        printf("new rootroot = %le ",root_dot);

/* add assumption */
/*
        pfimg = fimg;
        pfnoise = fnoise;
        pfnoise_tmp = fnoise_tmp;
        noise_mag = 0.0;
        for(j=0;j<patch_ydim;j++){
```

```c
                                for(k=0;k<patch_xdim;k++){
                                        dtemp = (double)*pfimg + (double)*pfnoise *
        INCREMENT;

                                        if( fabs( dtemp ) < 1.0 )*pfnoise_tmp = *pfnoise;
                                        else {
                                                if( dtemp >= 0.0){
                                                        *pfnoise_tmp = *pfnoise / (float)sqrt(
        dtemp );

                                                }
                                                else {
                                                        *pfnoise_tmp = *pfnoise / (float)sqrt( -
        dtemp );

                                                }
                                        }
                                        noise_mag += (double) *(pfnoise_tmp) * (double)
        *(pfnoise_tmp);

                                        pfnoise++; pfimg++;pfnoise_tmp++;
                                }
                        }
                        noise_mag = sqrt(noise_mag);
                        root_dot1 =
        dot_root_product(fimg,fnoise_tmp,patch_xdim*patch_ydim);
                        root_dot1 /= (noise_mag * img_mag_root);
                        printf("new root add =  %le  ",root_dot1);
        */

        /* subtract assumption */
        /*

                        pfimg = fimg;
                        pfnoise = fnoise;
                        pfnoise_tmp = fnoise_tmp;
                        noise_mag = 0.0;
                        for(j=0;j<patch_ydim;j++){
                                for(k=0;k<patch_xdim;k++){
                                        dtemp = (double)*pfimg - (double)*pfnoise *
        INCREMENT;

                                        if( fabs( dtemp ) < 1.0 )*pfnoise_tmp = *pfnoise;
                                        else {
                                                if( dtemp >= 0.0){
                                                        *pfnoise_tmp = *pfnoise / (float)sqrt(
        dtemp );

                                                }
                                                else {
                                                        *pfnoise_tmp = *pfnoise / (float)sqrt( -
        dtemp );

                                                }
```

13

```c
                                }
                                noise_mag += (double) *(pfnoise_tmp) * (double)
*(pfnoise_tmp);

                                pfnoise++; pfimg++;pfnoise_tmp++;
                        }
                }
                noise_mag = sqrt(noise_mag);
                root_dot2 =
dot_root_product(fimg,fnoise_tmp,patch_xdim*patch_ydim);
                root_dot2 /= (noise_mag * img_mag_root);
                printf("new root sub =  %le \n",root_dot2);
                printf("\ndiff1 = %le     diff2 = %le ",root_dot1 - root_dot,root_dot2-
root_dot);
*/


                if( dot > 0.0 ){
                        bit[i] = 1;
                        code_number+=value;
                }
                else {
                        bit[i] = 0;
                }
                printf("            bit value %d =   %d \n",i,bit[i]);
                value/=2;


        }
        printf("\n\nCode Number Found = %d\n\n",code_number);
        rms/=(double)(xdim*ydim);
        rms = sqrt(rms);
        printf("\n\nrms  = %le\n\n",rms);


        /* free and clean up */
        free(img);
        free(original);
        free(noise);
        free(fnoise);
        free(fnoise_tmp);
        free(fimg);
        free(foriginal);
        free(ar);
        free(ai);

}
```

```
/* this program takes in a raw 8 bit image which is in general a sub-section of
another original image. The orginal image will have already been signed. This
program needs to know what offest and extent that the sub-section has realtive to
the dimensions of the original. It also needs to have the master_noise file present
keyed to the original extent of the image also */

/* this one now uses the dot product approach, removing the innate bias from the
original */


#include "pinecone.h"
#include "../../sgi_tools/image.h"

#define BITS 16
#define VALUE 32768
#define INCREMENT 0.5
#define KCONST 500.0
#define ZERO_POINT 0.2

int bit[BITS];
long dim,bits,fftdim,fft_size;
float *ar,*ai;
double wr[10000],wi[10000];

int helplines = 7;
char    *help[] =
{
        "usage: recognize filename original_file orig_xdim orig_ydim x_offset y_offset
patch_xdim patch_ydim\n",
        "\n Program requires that the file 'master_key' exists in current directory",
        "\n filename is image in 8-bit raw form which is in general sub-section of
original image",
        "\n original_file is original image in 8-bit raw form ",
        "\n orig_xdim and orig_ydim is the pixel extent of the original image",
        "\n x_offset and y_offset is the offset in pixels of the upper left edge of the
sub-section rel. to original",
        "\n patch_xdim and patch_ydim is the pixel extent of the image patch being
recognized"
};




int general_purpose_filter(float *image, long xdim, long ydim, long which){
long i,j;
float *par,*pai,*pan,*pan_im,*pimage;
float *par1,*pai1,*pan1,*pan_im1;
```

Appendix D

```c
float k_value,ii,jj,ftemp,ftemp1,ratio;
float std_dev, total;
static float patch_std_dev;
static float original_std_dev;

memset(ar,0, fft_size * sizeof(float) );
memset(ai,0, fft_size * sizeof(float) );

/* copy image buffer into ar */
par = ar;
pimage = image;
for(i=0;i<ydim;i++){
        for(j=0;j<xdim;j++){
                *(par++) = *(pimage++);
        }
        par += (fftdim - xdim);
}

/* real 2d fft, in place */
realfft2d(ar,ai,bits,0,wr,wi);

/* apply low pass to everything */
        par = ar;pai=ai;
        pan = &ar[fftdim-1];pan_im=&ai[fftdim-1];
        par1 = &ar[fft_size - fftdim];pai1=&ai[fft_size-fftdim];
        pan1 = &ar[fft_size-1];pan_im1=&ai[fft_size-1];
        for(i=0;i<(1+fftdim/2);i++){
                ii = (float)i / (float)(fftdim/2);
                ii *= ii;
                for(j=0;j<(1+fftdim/2);j++){
                        jj = (float)j / (float)(fftdim/2);
                        jj = (float)sqrt( (double)(ii+jj*jj) );
                        if( jj > ZERO_POINT )jj =0.0;
                        else {
                                jj = (ZERO_POINT - jj)/ZERO_POINT;
                        }

                        *par *= jj;
                        *pai *= jj;
                        par++; pai++;
                        if(j!=0 && j!=(fftdim/2)){
                                *pan *= jj;
                                *pan_im *= jj;
                                pan--; pan_im--;
                        }
                }
```

```c
        par += (fftdim/2 -1); pai += (fftdim/2 -1);
        pan += (fftdim +fftdim/2 -1); pan_im += (fftdim+fftdim/2 -1);

    if(i!=0 && i!=(fftdim/2)){
            for(j=0;j<(1+fftdim/2);j++){
                    jj = (float)j / (float)(fftdim/2);
                    jj = (float)sqrt( (double)(ii+jj*jj) );
                    if( jj > ZERO_POINT )jj =0.0;
                    else {
                            jj = (ZERO_POINT - jj)/ZERO_POINT;
                    }

                    *(par1++) *= jj;
                    *(pai1++) *= jj;
                    if(j!=0 && j!=(fftdim/2)){
                            *(pan1--) *= jj;
                            *(pan_im1--) *= jj;
                    }
            }
            par1 -= (3*fftdim/2 +1); pai1 -= (3*fftdim/2 +1);
            pan1 -= (fftdim/2 +1); pan_im1 -= (fftdim/2 +1);
    }
}

/* if which == 0 or 1, find the standard deviation in frequency space */
if(which == 0 || which == 1){
    par = ar; pai = ai;
    pan = &ar[fftdim-1];pan_im=&ai[fftdim-1];
    std_dev = 0.0;
    total = 0.0;
    for(i=0;i<(1+fftdim/2);i++){
            ii = (float)i / (float)(fftdim/2);
            ii *= ii;
            for(j=0;j<(1+fftdim/2);j++){
                    jj = (float)j / (float)(fftdim/2);
                    jj = (float)sqrt((double)(ii+jj*jj));

                    ftemp = *par * *par + *pai * *pai;
                    std_dev += jj*ftemp;
                    total += ftemp;
                    par++; pai++;
                    if(j!=0 && j!=(fftdim/2)){
                            ftemp = *pan * *pan + *pan_im * *pan_im;
                            total += ftemp;
                            std_dev += jj*ftemp;
                            pan--; pan_im--;
```

```c
                }

            }
            par += (fftdim/2 -1); pai += (fftdim/2 -1);
            pan += (fftdim +fftdim/2 -1); pan_im += (fftdim+fftdim/2 -1);
        }
        std_dev /= total;
        printf("\n Std deviation found = %e ",std_dev);
    }

    if(which == 0) patch_std_dev = std_dev;
    if(which == 1) original_std_dev = std_dev;

    /* now filter it in the case of which == 1 */
    /* and in the case of which == 2, then use the same filter used in the case 1 */
    if( (which == 1 || which == 2)  && (original_std_dev > 0.95 * patch_std_dev) ){
        ratio = patch_std_dev / original_std_dev;
        ratio ;
        if(ratio > 1.0)k_value = 0.0;
        else {
            k_value = -KCONST * (1.0 - ratio) ;
        }
        std_dev /= total;
        par = ar;pai=ai;
        pan = &ar[fftdim-1];pan_im=&ai[fftdim-1];
        par1 = &ar[fft_size - fftdim];pai1=&ai[fft_size-fftdim];
        pan1 = &ar[fft_size-1];pan_im1=&ai[fft_size-1];
        std_dev = 0.0;
        total = 0.0;
        for(i=0;i<(1+fftdim/2);i++){
            ii = (float)i / (float)(fftdim/2);
            ii *= ii;
            for(j=0;j<(1+fftdim/2);j++){
                jj = (float)j / (float)(fftdim/2);
                jj = ii+jj*jj;
                ftemp1 = (float)sqrt((double)jj );

                jj = fexp( jj * k_value);
                *par *= jj;
                *pai *= jj;
                ftemp = *par * *par + *pai * *pai;
                std_dev += ftemp1*ftemp;
                total += ftemp;
                par++; pai++;
                if(j!=0 && j!=(fftdim/2)){
                    *pan *= jj;
```

4

```c
                                    *pan_im *= jj;
                                    ftemp = *pan * *pan + *pan_im * *pan_im;
                                    std_dev += ftemp1*ftemp;
                                    total += ftemp;
                                    pan--; pan_im--;
                            }
                    }
                    par += (fftdim/2 -1); pai += (fftdim/2 -1);
                    pan += (fftdim +fftdim/2 -1); pan_im += (fftdim+fftdim/2 -1);

                    if(i!=0 && i!=(fftdim/2)){
                            for(j=0;j<(1+fftdim/2);j++){
                                    jj = (float)j / (float)(fftdim/2);
                                    jj = ii+jj*jj;

                                    jj = fexp( jj * k_value);
                                    *(par1++) *= jj;
                                    *(pai1++) *= jj;
                                    if(j!=0 && j!=(fftdim/2)){
                                            *(pan1--) *= jj;
                                            *(pan_im1--) *= jj;
                                    }
                            }
                            par1 -= (3*fftdim/2 +1); pai1 -= (3*fftdim/2 +1);
                            pan1 -= (fftdim/2 +1); pan_im1 -= (fftdim/2 +1);
                    }
            }
            std_dev /= total;
    if(which == 1) printf("\n Std deviation post-filtering = %e ",std_dev);
    }



    realfft2d( ar, ai, bits, 1, wr, wi) ;

    par = ar;
    pimage = image;
    for(i=0;i<ydim;i++){
            for(j=0;j<xdim;j++){
                    *(pimage++) = *(par++);
            }
            par += (fftdim - xdim);
    }

    return(1);
```

5

```
}



double dot_product(float *img1,float *img2,long size){
long i;
float *pimg1,*pimg2;
double result = 0.0;

pimg1 = img1;
pimg2 = img2;
        for(i=0;i<size;i++) result += ( (double) *(pimg1++) * (double) *(pimg2++) );

return(result);
}

double dot_squared_product(float *img1,float *img2,long size){
long i;
float *pimg1,*pimg2;
double result = 0.0;
double dtemp;

pimg1 = img1;
pimg2 = img2;
        for(i=0;i<size;i++){
                dtemp = (double)*(pimg1++);
                if(dtemp < 0.0){
                        dtemp*=dtemp;
                        dtemp = -dtemp;
                }
                else dtemp *= dtemp;
                result += ( dtemp * (double) *(pimg2++) );
        }


return(result);
}

double dot_root_product(float *img1,float *img2,long size){
long i;
float *pimg1,*pimg2;
double result = 0.0;
double dtemp;

pimg1 = img1;
```

```
pimg2 = img2;
        for(i=0;i<size;i++){
                dtemp = (double)*(pimg1++);
                if(dtemp <0){
                        dtemp = sqrt(-dtemp);
                        dtemp = -dtemp;
                }
                else dtemp = sqrt(dtemp);
                result += ( dtemp  * (double)*(pimg2++) );
        }
return(result);
}

double dot_rootroot_product(float *img1,float *img2,long size){
long i;
float *pimg1,*pimg2;
double result = 0.0;
double dtemp;

pimg1 = img1;
pimg2 = img2;
        for(i=0;i<size;i++){
                dtemp = (double)*(pimg1++);
                if(dtemp <0){
                        dtemp = sqrt(-dtemp);
                        dtemp = sqrt(dtemp);
                        dtemp = -dtemp;
                }
                else {
                        dtemp = sqrt(dtemp);
                        dtemp = sqrt(dtemp);
                }
                result += ( dtemp  * (double)*(pimg2++) );
        }
return(result);
}


main( argc, argv )
int     argc ;
char    *argv[] ;
{
        long i,j,k,xdim,ydim,temp,patch_xdim,patch_ydim,xoffset,yoffset,index;
        long value = VALUE,code_number = 0,go;
        unsigned char tmp,*noise,*pnoise,*original,*poriginal,*p1;
        unsigned char *img,*pimg,*pimg1,stmp;
```

7

```c
char string[80], outfile[80];
FILE *inf;
float *fnoise,*pfnoise,total,*fimg,*pfimg,*foriginal,*pforiginal;
float *fnoise_tmp,*pfnoise_tmp;
double base_dot,dot,img_mag,original_mag,noise_mag,rms;
double square_dot,root_dot,rootroot_dot;
double base_square_dot,base_root_dot,dtemp;
double img_mag_root,img_mag_square,img_mag_rootroot;
double dot1,root_dot1,root_dot2;

if(argc != 9) {
        for(j=0;j<helplines;j++)fprintf( stderr, "%s", help[j]) ;
        exit( 1 ) ;
}
xdim = atoi( argv[3] );
ydim = atoi( argv[4] );
xoffset = atoi( argv[5] );
yoffset = atoi( argv[6] );
patch_xdim = atoi( argv[7] );
patch_ydim = atoi( argv[8] );

/* find the next power of two equal to or higher than patch */
if(patch_xdim > patch_ydim)dim = patch_xdim;
else dim = patch_ydim;
fftdim = 1; go = 1; bits = 0;
while( go ){
        if( dim > fftdim ){
                fftdim*=2;
                bits++;
        }
        else go = 0;
}
if(bits > 12){
        fprintf( stderr, "recognize : sorry, this particular program only accepts
4K images and less\n" ) ;
        exit( 1 ) ;
}
fft_size = fftdim * fftdim;

img = calloc(patch_xdim*patch_ydim, sizeof(unsigned char) ) ;
fimg = calloc(patch_xdim*patch_ydim, sizeof(float) ) ;
original = calloc(xdim*ydim, sizeof(unsigned char) ) ;
foriginal = calloc(patch_xdim*patch_ydim, sizeof(float) ) ;
noise = calloc(xdim*ydim, sizeof(unsigned char) ) ;
fnoise = calloc(patch_xdim*patch_ydim, sizeof(float) ) ;
fnoise_tmp = calloc(patch_xdim*patch_ydim, sizeof(float) ) ;
```

8

```c
ar = calloc(fft_size, sizeof(float) ) ;
ai = calloc(fft_size, sizeof(float) ) ;
if( !img || !noise || !original || !fnoise || !fnoise_tmp || !fimg || !foriginal || !ar ||
!ai){
        fprintf( stderr, "sign_it : can not allocate space\n" ) ;
        exit( 1 ) ;
}


/* read in Suspected Patch into img */
inf = fopen(argv[1],"r");
if(!inf) {
        fprintf(stderr,"sign_it: can't open %s\n",argv[1]);
        exit(1);
}
fread( img,sizeof(unsigned char),patch_xdim*patch_ydim,inf);
fclose(inf);

/* flip it */
pimg = img;
p1 = &img[patch_xdim*(patch_ydim-1)];
for(i=0;i<(patch_ydim/2);i++){
        for(j=0;j<patch_xdim;j++){
                tmp = *pimg;
                *(pimg++) = *p1;
                *(p1++) = tmp;
        }
        p1 -= (2*patch_xdim);
}

/* copy it into float array, removing mean value */
pimg = img;
total = 0.0;
for(i=0;i<(patch_ydim*patch_xdim);i++)total += (float) *(pimg++);
total /= (float)(patch_xdim*patch_ydim);
pimg = img;
pfimg = fimg;
for(i=0;i<(patch_ydim*patch_xdim);i++) *(pfimg++) = (float) *(pimg++) - total;
pfimg = fimg;
img_mag = 0.0;
for(i=0;i<(patch_ydim*patch_xdim);i++){
        dtemp = (double)*(pfimg++);
        dtemp *= dtemp;
        img_mag += dtemp;
}
img_mag = sqrt(img_mag);
```

```c
printf("img_mag = %lf ",img_mag);

        /* now find its std deviation in frequency space using the general purpose
filter function */
        general_purpose_filter(fimg,patch_xdim,patch_ydim,0);

        /* read in Original Unsigned Image */
        inf = fopen(argv[2],"r");
        if(!inf) {
                fprintf(stderr,"sign_it: can't open %s\n",argv[2]);
                exit(1);
        }
        fread(original,sizeof(unsigned char),xdim*ydim,inf);
        fclose(inf);

        /* create original patch */
        index = yoffset * xdim + xoffset;
        poriginal = original;
        p1 = &original[index];
        for(j=0;j<patch_ydim;j++){
                for(k=0;k<patch_xdim;k++){
                        *(poriginal++) = *(p1++);
                }
                p1 += (xdim - patch_xdim);
        }

        /* flip it */
        poriginal = original;
        p1 = &original[patch_xdim*(patch_ydim-1)];
        for(i=0;i<(patch_ydim/2);i++){
                for(j=0;j<patch_xdim;j++){
                        tmp = *poriginal;
                        *(poriginal++) = *p1;
                        *(p1++) = tmp;
                }
                p1 -= (2*patch_xdim);
        }

        /* copy it into float array, removing mean */
        poriginal = original;
        total = 0.0;
        for(i=0;i<(patch_ydim*patch_xdim);i++)total += (float) *(poriginal++);
        total /= (float)(patch_xdim*patch_ydim);
        poriginal = original;
        pforiginal = foriginal;
```

```
        for(i=0;i<(patch_ydim*patch_xdim);i++) *(pforiginal++) = (float) *(poriginal++) -
total;

        /* now find its std deviation in frequency space, then filter it to roughly match
the suspect patch */
        general_purpose_filter(foriginal,patch_xdim,patch_ydim,1);
        pforiginal = foriginal;
        original_mag = 0.0;
        for(i=0;i<(patch_ydim*patch_xdim);i++){
                dtemp = (double)*(pforiginal++);
                dtemp *= dtemp;
                original_mag += dtemp;
        }
        original_mag = sqrt(original_mag);
        printf("original_mag = %lf \n",original_mag);



        /* now remove the original image from img by using the dot product, or
whatever */
        dot = dot_product(foriginal,fimg,patch_xdim*patch_ydim);
printf(" initial dot = %le   ",dot/(original_mag*img_mag));
        dot /= (original_mag * original_mag);
printf("dot found = %le ",dot);
        pforiginal = foriginal;
        pfimg = fimg;
        img_mag = 0.0;
        for(i=0;i<(patch_ydim*patch_xdim);i++){
                *pfimg -= (dot * *(pforiginal++));

                /* now re-calculate the magnitudes */
                if( *pfimg < 0.0 ) dtemp = sqrt( -(double)*pfimg  );
                else dtemp = sqrt( (double)*pfimg );
                img_mag_rootroot += dtemp;
                dtemp = (double)*(pfimg++);
                if(dtemp < 0.0) img_mag_root -= dtemp;
                else img_mag_root += dtemp;
                dtemp *= dtemp;
                img_mag += dtemp;
                dtemp *= dtemp;
                img_mag_square += dtemp;
        }
        img_mag = sqrt(img_mag);
        img_mag_rootroot = sqrt(img_mag_rootroot);
        img_mag_root = sqrt(img_mag_root);
        img_mag_square = sqrt(img_mag_square);
```

11

```c
        dot = dot_product(foriginal,fimg,patch_xdim*patch_ydim);
printf(" new dot = %le  \n",dot/(original_mag*img_mag));



        /* open and read in master noise frame */
        inf = fopen("master_key","r");
        if(!inf) {
                fprintf(stderr,"sign_it: can't open %s\n","master_key");
                exit(1);
        }
        fread(noise,sizeof(unsigned char),xdim*ydim,inf);
        fclose(inf);
        yoffset = ydim - (yoffset + patch_ydim);  /* needed for flipping */



        /* main recoginition loop */

        rms = 0.0;
        for(i=0;i<BITS;i++){

                /* create current bit plane noise patch, fnoise */
                /* THE MEAN OF THESE MASTER NOISE PLANES SHOULD BE
ZERO!!! FROM THE SIGNING PROCESS */
                index = yoffset * xdim + xoffset;
                pnoise = &noise[index];
                pfnoise = fnoise;
                for(j=0;j<patch_ydim;j++){
                        for(k=0;k<patch_xdim;k++){
                                if( (index%BITS) == i ){
                                        *pfnoise = (float) (*pnoise - 128);
                                        rms += (double)*pfnoise * (double)*pfnoise;
                                }
                                else *pfnoise = 0.0;
                                pfnoise++;
                                pnoise++;
                                index++;
                        }
                        pnoise += (xdim - patch_xdim);
                        index += (xdim - patch_xdim);
                }

                /* now filter that patch using the same gaussian used on the original */
                general_purpose_filter(fnoise,patch_xdim,patch_ydim,2);
                pfnoise = fnoise;
```

```c
        noise_mag = 0.0;
        for(j=0;j<(patch_ydim*patch_xdim);j++){
                noise_mag += (double) *(pfnoise) * (double) *(pfnoise);
                pfnoise++;
        }
        noise_mag = sqrt(noise_mag);

        /* find dot product between suspect patch and filtered tmp noise */
        dot = dot_product(fnoise,fimg,patch_xdim*patch_ydim);
        dot /= (noise_mag * img_mag);

        printf("dot=%le ",dot);

/* new expiriment: re-weight the noise then do the root dot */
        pfimg = fimg;
        pfnoise = fnoise;
        noise_mag = 0.0;
        for(j=0;j<patch_ydim;j++){
                for(k=0;k<patch_xdim;k++){
                        dtemp = (double)*pfimg;
                        if( fabs( dtemp ) < 1.0 );
                        else {
                                if( dtemp >= 0.0){
                                        *pfnoise /= (float)sqrt( dtemp );
                                }
                                else {
                                        *pfnoise /= (float)sqrt( -dtemp );
                                }
                        }
                        noise_mag += (double) *(pfnoise) * (double) *(pfnoise);
                        pfnoise++; pfimg++;
                }
        }
        noise_mag = sqrt(noise_mag);
        root_dot = dot_root_product(fimg,fnoise,patch_xdim*patch_ydim);
        root_dot /= (noise_mag * img_mag_root);
        printf("   new root  = %le ",root_dot);

        pfimg = fimg;
        pfnoise = fnoise;
        pfnoise_tmp = fnoise_tmp;
        noise_mag = 0.0;
        for(j=0;j<patch_ydim;j++){
                for(k=0;k<patch_xdim;k++){
                        dtemp = (double)*pfimg;
                        if( fabs( dtemp ) < 1.0 )*pfnoise_tmp = *pfnoise;
```

14. 13

```c
                    else {
                        if( dtemp >= 0.0){
                            *pfnoise_tmp = *pfnoise / (float)sqrt(
dtemp );
                        }
                        else {
                            *pfnoise_tmp = *pfnoise / (float)sqrt( -
dtemp );
                        }
                    }
                    noise_mag += (double) *(pfnoise_tmp) * (double)
*(pfnoise_tmp);
                    pfnoise++; pfimg++;pfnoise_tmp++;
                }
            }
        noise_mag = sqrt(noise_mag);
        root_dot = dot_root_product(fimg,fnoise_tmp,patch_xdim*patch_ydim);
        root_dot /= (noise_mag * img_mag_root);
        printf("new rootroot  =  %le ",root_dot);

/* add assumption */
/*
        pfimg = fimg;
        pfnoise = fnoise;
        pfnoise_tmp = fnoise_tmp;
        noise_mag = 0.0;
        for(j=0;j<patch_ydim;j++){
            for(k=0;k<patch_xdim;k++){
                    dtemp = (double)*pfimg + (double)*pfnoise *
INCREMENT;
                    if( fabs( dtemp ) < 1.0 )*pfnoise_tmp = *pfnoise;
                    else {
                        if( dtemp >= 0.0){
                            *pfnoise_tmp = *pfnoise / (float)sqrt(
dtemp );
                        }
                        else {
                            *pfnoise_tmp = *pfnoise / (float)sqrt( -
dtemp );
                        }
                    }
                    noise_mag += (double) *(pfnoise_tmp) * (double)
*(pfnoise_tmp);
                    pfnoise++; pfimg++;pfnoise_tmp++;
                }
            }
```

```c
                noise_mag = sqrt(noise_mag);
                root_dot1 =
dot_root_product(fimg,fnoise_tmp,patch_xdim*patch_ydim);
                root_dot1 /= (noise_mag * img_mag_root);
                printf("new root add = %le  ",root_dot1);
*/

/* subtract assumption */
/*
                pfimg = fimg;
                pfnoise = fnoise;
                pfnoise_tmp = fnoise_tmp;
                noise_mag = 0.0;
                for(j=0;j<patch_ydim;j++){
                        for(k=0;k<patch_xdim;k++){
                                dtemp = (double)*pfimg - (double)*pfnoise *
INCREMENT;

                                if( fabs( dtemp ) < 1.0 )*pfnoise_tmp = *pfnoise;
                                else {
                                        if( dtemp >= 0.0){
                                                *pfnoise_tmp = *pfnoise / (float)sqrt(
dtemp );

                                        }
                                        else {
                                                *pfnoise_tmp = *pfnoise / (float)sqrt( -
dtemp );

                                        }
                                }
                                noise_mag += (double) *(pfnoise_tmp) * (double)
*(pfnoise_tmp);
                                pfnoise++; pfimg++;pfnoise_tmp++;
                        }
                }
                noise_mag = sqrt(noise_mag);
                root_dot2 =
dot_root_product(fimg,fnoise_tmp,patch_xdim*patch_ydim);
                root_dot2 /= (noise_mag * img_mag_root);
                printf("new root sub = %le \n",root_dot2);
                printf("\ndiff1 = %le     diff2 = %le ",root_dot1 - root_dot,root_dot2-
root_dot);
*/


                if( dot > 0.0 ){
                        bit[i] = 1;
                        code_number+=value;
```

```c
        }
        else {
                bit[i] = 0;
        }
        printf("            bit value %d  =   %d \n",i,bit[i]);
        value/=2;


}
printf("\n\nCode Number Found = %d\n\n",code_number);
rms/=(double)(xdim*ydim);
rms = sqrt(rms);
printf("\n\nrms  = %le\n\n",rms);


/* free and clean up */
free(img);
free(original);
free(noise);
free(fnoise);
free(fnoise_tmp);
free(fimg);
free(foriginal);
free(ar);
free(ai);

}
```

U.S. District Court

U.S. District Court of Eastern Pennsylvania (Philadelphia)

CIVIL DOCKET FOR CASE #: 01-CV-5497

AIRCLIC INC. v. NEOMEDIA TECHNOLOGIE

Filed: 10/30/01
Assigned to: JUDGE LOUIS H. POLLAK
Demand: $0,000
Nature of Suit: 830
Lead Docket: None
Jurisdiction: Federal Question
Dkt# in other court: None
Cause: 35:271 Patent Infringement

AIRCLIC INC. HOWARD D. SCHER
PLAINTIFF [COR LD NTC]
BUCHANAN INGERSOLL PROFESSIONAL
CORP
ELEVEN PENN CENTER, 14TH FL.
1835 MARKET ST.
PHILADELPHIA, PA 19103
USA
TEL 215-665-8700
v.
NEOMEDIA TECHNOLOGIES, INC. A. CHRISTOPHER YOUNG
DEFENDANT [COR LD NTC]
PEPPER HAMILTON, LLP
18TH & ARCH STS.
3000 TWO LOGAN SQUARE
PHILA, PA 19103-2799
USA
TEL 215-981-4750

DOCKET PROCEEDINGS

DATE # DOCKET ENTRY

10/30/01 1 Complaint. filing fee $ 150 receipt # 774975 (ss)
[Entry date 10/31/01]

10/30/01 -- Summons(es) issued, One original given to: counsel

10/30/01
(ss) [Entry date 10/31/01]

10/30/01 -- Standard Case Management Track. (ss) [Entry date
10/31/01]

10/30/01 2 Copy of Form to Commissioner of Patent. (ss)
[Entry date 10/31/01]

11/6/01 3 Affidavit of: RAYMOND GIVIGLIANE re: served summons and
complaint upon DEFENDANT NEOMEDIA TECHNOLOGIE . Served
by:
PERSONAL SERVICE on 11/1/01 . (jl) [Entry date 11/06/01]

11/21/01 4 MOTION by DEFENDANT NEOMEDIA TECHNOLOGIE TO DISMISS THE
COMPLAINT PURSUANT TO FED.R.CIV.P.12 (b)(1) AND (3) ,
MEMORANDUM, CERTIFICATE OF SERVICE. (jl)
[Entry date 11/21/01]

12/10/01 5 STIPULATION AND ORDER ENLARGING TIME THAT PLAINTIFF SHALL
HAVE UNTIL 12/19/01 TO FILE A REPLY AND OPPOSITION TO
DEFENDANT'S MOTION TO DISMISS; DEFENDANT SHALL HAVE UNTIL
12/27/01 TO FILE A REPLY MEMORANDUM IN SUPPORT OF ITS
MOTION TO DISMISS (SIGNED BY JUDGE LOUIS H. POLLAK)
12/10/01 ENTERED AND COPIES MAILED. (np)
[Entry date 12/10/01]

12/19/01 6 Memorandum by PLAINTIFF AIRCLIC INC. in opposition to
DEFENDANT NEOMEDIA TECHNOLOGIES, INC.'S MOTION TO DISMISS
THE COMPLAINT PURSUANT TO FED.R.CIV.P.12 (b)(1) AND (3) ,
Certificate of Service. (jl) [Entry date 12/20/01]

12/27/01 7 Reply Memorandum of law in further support of Defendant
NEOMEDIA TECHNOLOGIES, INC.'S motion to dismiss the
complaint pursuant to Federal rule of civil procedure 12
(b)(1) and (3) , Certificate of Service. (jl)
[Entry date 12/27/01]

1/7/02 8 MOTION by PLAINTIFF AIRCLIC INC. FOR LEAVE TO FILE
SUR-REPLY MEMORANDUM IN OPPOSITION TO DEFENDANT NEOMEDIA
TECHNOLOGIES, INC.'S MOTION TO DISMISS THE COMPLAINT
PURSUANT TO FED.R.CIV.P. 12 (B) (1) AND (3), CERTIFICATE
OF SERVICE. (sur-reply attached) (jl) [Entry date
01/08/02]

1/16/02 9 ORDER THAT PLAINTIFF'S AIRCLIC MOTION FOR LEAVE TO FILE
SUR-REPLY MEMORANDUM IN OPPOSITION TO DEFENDANT NEOMEDIA
INC.'S MOTION TO DISMISS THE COMPLAINT PURSUANT TO
FED.R.CIV.P. 12 (B) (1) AND (3) IS GRANTED. THE SUR-REPLY

MEMORANDUM ATTACHED TO PLAINTIFF'S MOTION SHALL BE DEEMED
FILED WITH THE CLERK OF COURT AS OF THE DATE OF THIS
ORDER
(SIGNED BY JUDGE LOUIS H. POLLAK) 1/16/01 ENTERED AND
COPIES MAILED. (jl) [Entry date 01/16/02]

1/16/02 10 Sur-Reply Memorandum by PLAINTIFF AIRCLIC INC. in
opposition to DEFENDANT NEOMEDIA TECHNOLOGIES MOTION TO
DISMISS THE COMPLAINT PURSUANT TO FED.R.CIV.P.12 (b)(1)
AND
(3). (jl) [Entry date 01/17/02]